

3575/ITS/H/91 ✓

# IMPLEMENTASI PROSESOR SINYAL DIGITAL TMS32010 UNTUK PENGATURAN POSISI MENGUNAKAN MOTOR DC



PSE  
Gal. 391 6  
Sul  
I-1  
1990

Oleh :

**S U L Y A N T O**

NRP. 2852200251


JURUSAN TEKNIK ELEKTRO  
FAKULTAS TEKNOLOGI INDUSTRI  
INSTITUT TEKNOLOGI SEPULUH NOPEMBER  
S U R A B A Y A  
1 9 9 0

# **IMPLEMENTASI PROSESOR SINYAL DIGITAL TMS32010 UNTUK PENGATURAN POSISI MENGUNAKAN MOTOR DC**

## **TUGAS AKHIR**

**Diajukan Guna Memenuhi Sebagian Persyaratan  
Untuk Memperoleh Gelar  
Sarjana Teknik Elektro  
pada  
Bidang Studi Elektronika  
Di  
Jurusan Teknik Elektro  
Fakultas Teknologi Industri  
Institut Teknologi Sepuluh Nopember  
Surabaya**

**Mengetahui / Menyetujui  
Dosen Pembimbing**



**( DR. Ir. Handayani Tjandrasa MSc )**

**S u r a b a y a**

**J u l i    1990**

## ABSTRAK

Dengan pesatnya kemajuan di bidang teknologi elektronika khususnya elektronika digital dan mikroprosesor, maka sistem kontrol banyak beralih dari sistem kontrol analog ke sistem kontrol digital terutama menggunakan mikroprosesor/ mikrokontroler sebagai unit pengolah yang mempunyai keandalan tinggi.

Kontroler PID digital adalah salah satu contohnya yang merupakan alternatif penggunaan kontroler PID analog. Pada kontroler PID digital terdapat beberapa keunggulan dibandingkan dengan kontroler PID analog antara lain: koefisien-koefisiennya dapat diprogram sehingga penggunaannya menjadi fleksibel, parameter kontroler tidak berubah oleh pengaruh suhu, waktu dan keadaan sekitarnya.

Pada tugas akhir ini direncanakan suatu sistem kontroler PID digital dengan mengimplementasikan prosesor sinyal digital TMS32010 sebagai unit pengolah utama. TMS32010 adalah suatu prosesor khusus untuk aplikasi pengolahan sinyal digital yang mempunyai kecepatan eksekusi serta operasi aritmatik yang sangat tinggi sehingga output sistem akan bersifat real-time. Output sistem ini nantinya digunakan untuk mengatur posisi sudut putar motor DC.

Program yang diperlukan untuk menjalankan prosesor ini serta koefisien-koefisiennya ditransfer dari komputer IBM PC/XT dengan bantuan bahasa tingkat tinggi Turbo Pascal.

## **KATA PENGANTAR**

Puji syukur penulis panjatkan kepada Tuhan Yang Maha Esa atas segala rahmat dan petunjuk-Nya sehingga penulis dapat merampungkan tugas akhir yang berjudul:

### **IMPLEMENTASI PROSESOR SINYAL DIGITAL**

#### **TMS32010 UNTUK PENGATURAN POSISI**

#### **MENGGUNAKAN MOTOR DC**

Tugas akhir ini mempunyai beban 6 SKS (satuan kredit semester) yang disusun untuk memenuhi sebagian persyaratan untuk menyelesaikan studi program S<sub>1</sub> pada Jurusan Teknik Elektro Fakultas Teknologi Industri Institut Teknologi Sepuluh Nopember Surabaya.

Dalam mengerjakan tugas akhir ini penulis mengacu pada teori-teori yang telah didapat selama kuliah, dari literatur dan juga tidak lepas dari arahan dosen pembimbing yang telah banyak membantu dari awal hingga selesainya pengerjaan tugas akhir ini.

Penulis berharap bahwa tugas akhir ini akan bermanfaat bagi para pembaca serta dapat lebih disempurnakan dan dikembangkan dengan semaksimal mungkin.

Akhirnya tidak lupa penulis mengucapkan banyak terima kasih kepada:

- Ibu DR. Ir. Handayani Tj.,Msc. selaku Dosen Pembimbing, Dosen Wali dan Koordinator Bidang Studi Elektronika.

- Bapak-bapak dosen, khususnya dosen pada bidang studi Elektronika yang telah memberikan bekal ilmu selama penulis menempuh kuliah.
- Semua pihak yang telah membantu dalam menyelesaikan tugas akhir ini.

Surabaya, Juli 1990

Penulis

## DAFTAR ISI

ABSTRAK.....	i
KATA PENGANTAR.....	ii
DAFTAR ISI.....	iv
DAFTAR GAMBAR.....	viii
DAFTAR TABEL.....	xi

### BAB

I.	PENDAHULUAN.....	1
	I.1 LATAR BELAKANG.....	1
	I.2 PERMASALAHAN.....	1
	I.3 PEMBATAHAN PERMASALAHAN.....	2
	I.4 METODOLOGI.....	2
	I.5 LANGKAH-LANGKAH PEMBAHASAN.....	3
II.	TEORI PENUNJANG.....	5
	II.1 PENDAHULUAN.....	5
	II.2 KONTROLER PID DIGITAL.....	5
	II.2.1 Desain Kontroler PID Digital.....	6
	II.2.2 Realisasi Fungsi Alih Pulsa.....	15
	II.2.3 Realisasi Fungsi Alih Kontroler PID Digital.....	18
	II.3 MOTOR ARUS SEARAH (DC).....	20
	II.3.1 Prinsip Dasar Motor DC.....	21
	II.3.2 Konstruksi Motor DC.....	21
	II.3.3 Torsi Motor.....	23
	II.3.4 Hubungan-hubungan yang Berlaku Pada Motor DC.....	24

II.3.5	Jenis-jenis Motor DC.....	25
II.3.6	Motor DC Sebagai Motor Servo.....	27
II.4	KONVERTER DATA.....	29
II.4.1	Rangkaian Dasar DAC.....	30
II.4.2	Rangkaian Dasar ADC.....	33
III.	MIKROPROSESOR/MIKROKOMPUTER TMS32010.....	37
III.1	DESKRIPSI UMUM.....	37
III.2	ARSITEKTUR TMS32010.....	41
III.3	ORGANISASI MEMORI.....	46
III.3.1	Memori Data.....	47
III.3.2	Memori Program.....	48
III.3.3	Perpindahan Data.....	48
III.3.4	Pemetaan Memori.....	49
III.3.5	Auxiliary Register.....	50
III.3.6	Mode Memory Addressing.....	52
III.4	CENTRAL ARITHMETIC LOGIC UNIT (CALU)..	53
III.4.1	Shifter.....	54
III.4.2	ALU dan Accumulator.....	56
III.4.3	Multiplier, T Register dan P Register.....	58
III.5	SISTEM KONTROL.....	59
III.5.1	Program Counter dan Stack.....	60
III.5.2	Reset.....	62
III.5.3	Register Status.....	62
III.6	FUNGSI INPUT/OUTPUT.....	65
III.6.1	Operasi Input/Output.....	65
III.6.2	Operasi Table Read/Table Write...	67

III.6.3	Pin I/O Keperluan Umum (BIO).....	68
III.7	INTERRUPT.....	69
III.8	INSTRUKSI BAHASA ASSEMBLY TMS32010....	71
III.8.1	Mode Direct Addressing.....	72
III.8.2	Mode Indirect Addressing.....	73
III.8.3	Mode Immediate Addressing.....	74
III.8.4	Set Instruksi.....	75
IV.	PERENCANAAN.....	78
IV.1	PENDAHULUAN.....	78
IV.2	SLOT PERIPHERAL IBM PC/XT.....	80
IV.3	MODUL INTERFACE.....	86
IV.4	SISTEM KONTROLER.....	90
IV.4.1	Pemetaan Memori dan I/O.....	90
IV.4.2	Decoding.....	92
IV.4.3	Sistem Clock.....	94
IV.4.4	Sistem Reset.....	95
IV.5	RANGKAIAN I/O.....	96
IV.5.1	Transduser dan Konverter Analog ke Digital.....	96
IV.5.2	Konverter Digital ke Analog.....	98
IV.5.3	Rangkaian Pengaman dan Isolasi....	100
IV.6	RANGKAIAN AMPLIFIER.....	102
IV.6.1	Differensial Amplifier.....	102
IV.6.2	Power Amplifier.....	104
IV.6.2.1	Perencanaan Titik Kerja DC...	104
IV.6.2.2	Analisa AC.....	107
III.7	MOTOR.....	109



IV.8	PERENCANAAN PERANGKAT LUNAK(SOFTWARE).	109
IV.8.1	Routine Transfer Data.....	110
IV.8.2	Routine Pengaturan Motor oleh TMS32010.....	110
V.	PENGUJIAN ALAT.....	114
V.1	PENDAHULUAN.....	114
V.2	PENGUJIAN MODUL KONTROLER.....	114
V.3	PENGUJIAN MODUL SENSOR.....	117
V.3.1	Pengujian ADC 0804.....	117
V.3.2	Konversi Posisi Derajat ke Byte Data.....	118
V.4	MODUL AKTUATOR.....	120
V.4.1	Pengujian DAC 0808.....	120
V.4.2	Pengujian Differensial Amplifier...	120
V.4.3	Pengujian Power Amplifier.....	121
VI.	KESIMPULAN.....	124
	DAFTAR PUSTAKA.....	126
	LAMPIRAN.....	128

## DAFTAR GAMBAR

### GAMBAR

2-1	Diagram Blok Sistem Kontrol Otomatik.....	6
2-2	Kontroler PID.....	7
2-3	Respon Frekuensi Kontroler PI.....	8
2-4	Respon Frekuensi Kontroler PD.....	9
2-5	Respon Frekuensi Kontroler PID.....	10
2-6	Kontroler PID Digital.....	12
2-7	Sistem Kontrol Digital dengan Kontroler PID..	12
2-8	State Diagram untuk Persamaan (2.20).....	18
2-9	State Diagram untuk Persamaan (2.22).....	20
2-10	Bagian-bagian Motor DC.....	22
2-11	Torsi Putar pada Motor DC.....	23
2-12	Rangkaian Pengganti untuk Motor DC Seri.....	26
2-13	Rangkaian Pengganti untuk Motor DC Shunt.....	26
2-14	Rangkaian Pengganti untuk Motor DC Kompon....	26
2-15	Motor Servo DC.....	29
2-16	Rangkaian Binary Weighted Resistor DAC.....	31
2-17	R-2R Resistor Ladder DAC.....	32
2-18	Rangkaian Parallel Comparator/Flash ADC.....	34
2-19	Diagram Blok Successive Approximation ADC....	35
3-1	IC Prosesor TMS32010.....	38
3-2	Diagram Blok TMS32010.....	44
3-3	On-Chip Data RAM.....	47
3-4	Pemetaan Memori untuk TMS32010.....	49
3-5	Indirect Addressing Autoincrement.....	50

3-6	Indirect Addressing Autodecrement.....	51
3-7	Metoda-metoda Addressing pada TMS32010.....	52
3-8	Central Arithmetic Logic Unit (CALU).....	54
3-9	Arsitektur Harvard.....	61
3-10	Organisasi Register Status.....	63
3-11	Diagram Instruksi IN.....	66
3-12	Diagram Instruksi OUT.....	66
3-13	Diagram Instruksi TBLR.....	67
3-14	Diagram Instruksi TBLW.....	68
3-15	Diagram Waktu Interrupt.....	71
3-16	Format Instruksi Mode Direct Addressing.....	72
3-17	Format Instruksi Mode Indirect Addressing....	74
4-1	Diagram Blok Sistem Kontroler.....	79
4-2	Slot Peripheral IBM PC/XT.....	85
4-3	Pemetaan Memori Komputer IBM PC/XT.....	87
4-4	Rangkaian Decoder Address Memori Program.....	88
4-5	Rangkaian Isolasi TMS32010 dengan Memori Program.....	89
4-6	Rangkaian Pengubah Bus Data 8-bit ke 16-bit..	90
4-7	Pemetaan Memori dan I/O.....	91
4-8	Rangkaian Memori Program.....	93
4-9	Sistem Pendekodean I/O.....	94
4-10	Sistem Clock Internal.....	95
4-11	Rangkaian Reset TMS32010.....	96
4-12	Rangkaian Transduser dan ADC.....	97
4-13	Rangkaian Latch dan DAC.....	99
4-14	Rangkaian Pengaman dan Isolasi.....	101
4-15	Rangkaian Differensial Amplifier.....	102

4-16	Rangkaian Power Amplifier Kelas B.....	104
4-17	Rangkaian Ekvivalen DC.....	105
4-18	Diagram Alir Program Transfer Data.....	111
4-19	Diagram Alir Program Pengaturan Motor.....	113
5-1	Bentuk Fisik Alat yang dibuat.....	115
5-2	Frekuensi CLKOUT TMS32010.....	115
5-3	Strobe MEN dan DEN untuk Instruksi IN.....	116
5-4	Strobe MEN dan WE untuk Instruksi OUT.....	117
5-5	Clock pada ADC 0804.....	118
5-6	Sinyal Output INTR ADC 0804.....	119

## DAFTAR TABEL

### TABEL

3-1	Hasil-hasil Operasi Logika Accumulator.....	56
3-2	Definisi Bit-bit Register Status.....	64
3-3	Simbol-simbol Instruksi TMS32010.....	75
5-1	Konversi Posisi Derajat ke Data Byte.....	119
5-2	Hasil Konversi DAC.....	120
5-3	Pengukuran Differensial Amplifier.....	121
5-4	Pengukuran Power Amplifier.....	122
5-5	Hasil Pengamatan Ketelitian Posisi Derajat...	122

# **BAB I**

## **PENDAHULUAN**

### **I.1 LATAR BELAKANG**

Sistem kontrol otomatis sudah tidak asing lagi dalam dunia teknologi, baik teknologi elektronik maupun yang non-elektronik, karena kontrol otomatis memberikan kemudahan dalam pengaturan suatu sistem agar diperoleh performansi yang sesuai.

Dengan berkembangnya teknologi elektronika khususnya di bidang mikroelektronika dan digital, maka sistem kontrol otomatis mulai mengarah ke teknologi digital. Juga dengan lahirnya chip-chip mikroprosesor yang menawarkan banyak keandalan, merangsang para perancang untuk beralih ke sistem kontrol dengan mempergunakan mikroprosesor atau mikrokomputer.

Dalam sistem kontrol dikenal sistem pengaturan motor dengan kontroler PID (Proportional, Integral and Derivative). Salah satu aplikasinya adalah pada pengaturan posisi beban mekanik yang sesuai dengan posisi acuannya, atau dikenal dengan *servomekanisme posisi*. Dengan pengaturan ini diharapkan akan diperoleh keluaran sistem yang tepat, stabil, dan mempunyai respon yang cepat untuk mencapai kondisi yang dikehendaki.

### **I.2 PERMASALAHAN**

Pada sistem kontrol PID analog biasanya diimplementasikan dengan menggunakan penguat operasi (OP-AMP)

atau dengan rangkaian transistor diskrit. Dalam tugas akhir ini fungsi dari OP-AMP atau transistor tersebut digantikan oleh chip prosesor sinyal digital TMS32010 dengan beberapa perangkat tambahannya. Persamaan yang ada pada sistem kontrol PID ditulis dalam suatu program (software) sehingga secara keseluruhan alat ini dapat berfungsi sebagai pengganti sistem PID analog. Dengan cara di atas diharapkan respon sistem menjadi lebih baik dibandingkan dengan pengaturan secara analog.

### I.3 PEMBATASAN PERMASALAHAN

Titik berat pembahasan dalam tugas akhir ini adalah implementasi dari prosesor sinyal digital TMS32010, suatu chip prosesor khusus yang memiliki 16 bit bus data dengan kemampuan 4 K word program memori.

Prosesor tersebut bersama-sama dengan perangkat tambahan lainnya mengatur sudut putar motor DC sesuai dengan sudut referensi yang diberikan. Sebagai sensor posisi digunakan transduser putaran ke tegangan berupa potensiometer linier yang dikopel dengan poros motor.

### I.4 METODOLOGI

Sebagai langkah pertama dalam perencanaan sistem adalah menentukan diagram blok sistem pengaturan posisi yang sesuai. Dari diagram blok tersebut kemudian di-realisasikan ke bentuk rangkaian. Pada langkah ini akan

menghadapi banyak pilihan untuk menentukan tipe rangkaian maupun jenis komponen yang akan dipakai agar diperoleh spesifikasi yang sesuai.

Langkah selanjutnya, membuat fungsi alih sistem kontrol PID. Langkah ini dilakukan melalui cara pendekatan analog, yaitu menentukan fungsi alih dari rangkaian PID analog kemudian mengkonversikannya ke bentuk persamaan waktu diskrit untuk diimplementasikan pada mikroprosesor. Dalam hal ini fungsi alih dari motor diabaikan.

Dari fungsi alih yang didapat kemudian ditentukan koefisien-koefisien yang diperlukan. Penentuan koefisien ini dilakukan dengan cara mencoba-coba hingga diperoleh koefisien yang tepat, artinya keluaran dari sistem berupa sinyal redaman kritis(critical damp) dan tidak terjadi osilasi. Dengan bantuan komputer IBM PC/XT besarnya koefisien dan instruksi-instruksi untuk menjalankan prosesor TMS32010 diisikan ke dalam RAM statik yang berfungsi sebagai memori program melalui teknik *interface*.

## **I.5 LANGKAH-LANGKAH PEMBAHASAN**

Bab II : Pada bab ini akan dijelaskan mengenai teori yang menunjang dalam perencanaan dan pembuatan sistem pengaturan posisi dengan motor DC. Di sini meliputi teori sistem kontrol PID, prinsip kerja motor arus searah dan prinsip-prinsip pengaturannya, serta kom-



ponen-komponen penunjang lainnya.

Bab III : Pada bab III khusus akan dibahas mengenai prosesor sinyal digital TMS32010.

Bab IV : Pada bab ini akan dibahas langkah-langkah perencanaan dan pembuatan sistem pengaturan posisi dengan motor DC, meliputi sistem dekoding memori maupun I/O dari sistem yang akan dibuat, rangkaian ADC dan DAC, rangkaian transduser serta rangkaian penguat daya untuk penggerak motor.

Bab V : Bab ini berisi tentang pengujian sistem yang dibuat dan hasilnya dibandingkan dengan perencanaan semula.

Bab VI : Pada bagian ini akan diberikan kesimpulan dari keseluruhan perencanaan dan pembuatan sistem serta beberapa saran yang perlu.

## **BAB II**

### **TEORI PENUNJANG**

#### **II.1 PENDAHULUAN**

Dalam bab II ini akan dibahas tentang dasar-dasar teori yang menunjang dalam perencanaan sistem pengaturan posisi dengan motor DC yang mengimplementasikan prosesor sinyal digital sebagai komponen pengendali utama.

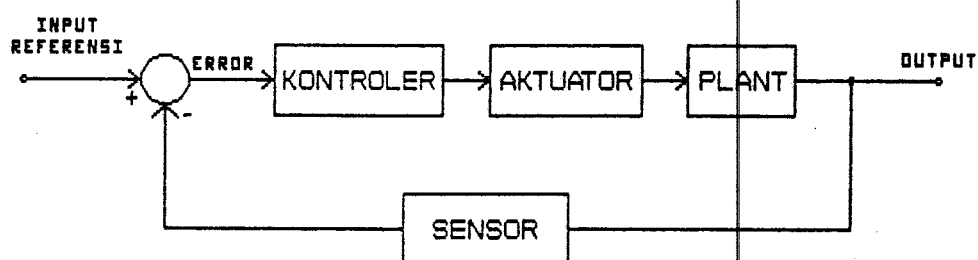
Mula-mula akan dibahas tentang prinsip kontrol PID dalam sistem analog serta pendekatannya ke bentuk digital untuk nantinya diimplementasikan pada prosesor dalam bentuk program. Dilanjutkan dengan pembahasan tentang prinsip kerja motor DC dan cara-cara pengaturannya. Dan terakhir akan dibahas mengenai proses konversi data yaitu konversi DAC dan ADC.

#### **II.2 KONTROLER PID DIGITAL**

Prinsip kontroler otomatis adalah membandingkan harga output sistem terhadap suatu harga referensi, menentukan deviasi dan menghasilkan suatu sinyal kontrol yang akan memperkecil deviasi sampai nol atau sampai suatu harga yang terkecil. Cara kontroler otomatis menghasilkan sinyal kontrol disebut sebagai aksi pengontrolan (control action).

Sistem kontrol otomatis pada umumnya terdiri dari empat bagian utama: sensor, kontroler, aktuator dan plant. Sensor mengukur perilaku dari sistem. Aktuator

memberikan daya untuk menggerakkan peralatan yang dikontrol (plant) agar mencapai suatu harga yang diinginkan. Sedangkan kontroler memberikan sinyal perintah ke aktuator sesuai dengan besarnya deviasi (error) yang diukur oleh sensor serta oleh aksi pengontrolan dari kontroler. Diagram blok sistem kontrol otomatis seperti pada gambar 2.1.



Gambar 2-1

#### DIAGRAM BLOK SISTEM KONTROL OTOMATIK

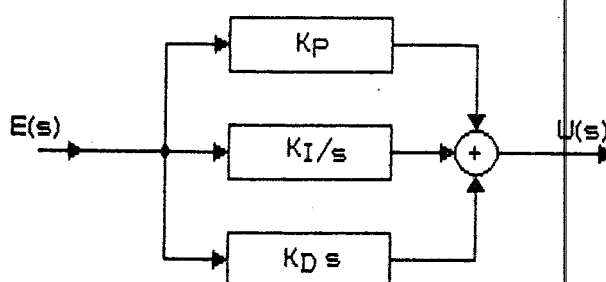
##### II.2.1 Desain Kontroler PID Digital

Dalam industri terdapat beberapa klasifikasi pengontrolan secara klasik sesuai dengan aksi pengontrolannya sebagai berikut:

1. Pengontrol dua posisi atau "on-off".
2. Pengontrol proporsional.
3. Pengontrol integral.
4. Pengontrol proporsional plus integral (PI).
5. Pengontrol proporsional plus turunan (PD).
6. Pengontrol proporsional plus integral plus turunan (PID).

Pada desain sistem kontrol analog (waktu kontinyu) banyak digunakan kontroler PID karena tipe ini mempunyai kestabilan dan performansi sistem yang baik. Diagram blok kontroler PID analog ditunjukkan pada gambar 2-2. Fungsi alih dari kontroler ini adalah:

$$\frac{U(s)}{E(s)} = D(s) = K_P + \frac{K_I}{s} + K_D s \quad (2.1)$$



Gambar 2-2<sup>1)</sup>

#### KONTROLER PID

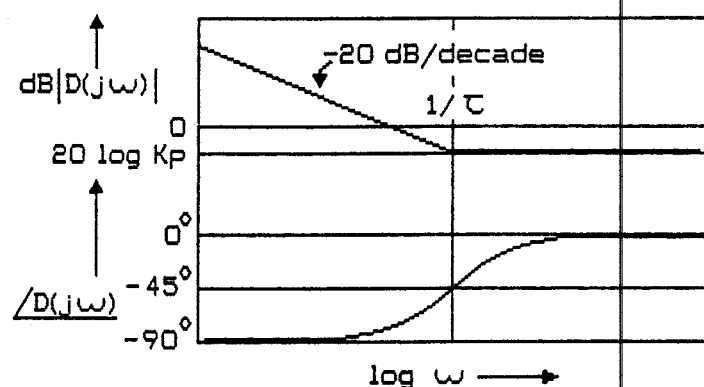
Kontroler PID adalah tipe lag-lead compensator. Pada pembahasan ini tidak akan dijelaskan mengenai compensator, namun dengan melihat gambar *Bode plot* yang diberikan di bawah, akan memberikan gambaran yang cukup jelas tentang hal ini.

Pertama akan dibahas kontroler PI (proportional-plus-integral). Fungsi alih dari kontroler PI adalah:

$$D(s) = K_P + \frac{K_I}{s} = \frac{K_P s + K_I}{s} = K_I \frac{1 + s\sigma}{s} ; \sigma = K_P/K_I$$

1) M. Gopal, 1988, Digital Control Engineering, p.137

Ini merupakan phase-lag compensator dengan pole terletak pada  $s=0$ . Respon frekuensi compensator ini ditunjukkan pada gambar 2-3.



Gambar 2-3<sup>2)</sup>

#### RESPON FREKUENSI KONTROLER PI

Dari Bode plot pada gambar 2-3 terlihat bahwa Kontroler PI meningkatkan penguatan pada frekuensi rendah (tak terhingga pada  $\omega=0$ ), dengan demikian mengurangi steady-state error. Jadi kontroler PI berfungsi sebagai phase-lag compensator.

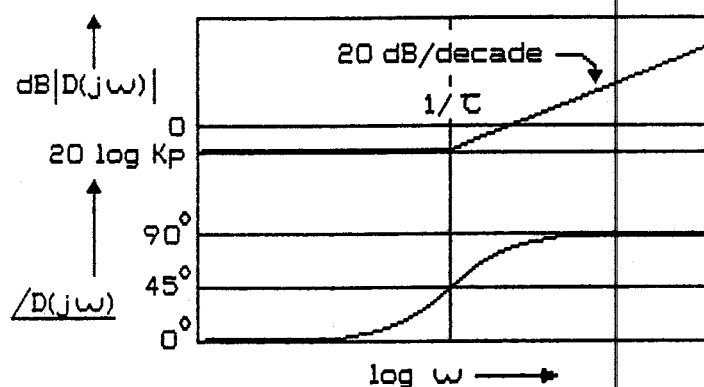
Berikutnya mengenai kontroler PD (proportional-plus-derivative). Fungsi alih dari kontroler ini adalah:

$$D(s) = K_p + K_D s = K_p(1 + \sigma s); \sigma = K_D/K_p$$

Ini adalah phase-lead compensator dengan pole terletak pada  $s=\infty$ . Respon frekuensi compensator ini ditunjukkan pada gambar 2-4.

2) loc. cit.

Kontroler PD menambah sudut phase positif pada respon frekuensi sistem, sehingga menambah kestabilan sistem dan meningkatkan bandwidth sistem. Pengaruh kontroler PD muncul pada frekuensi tinggi sedangkan pengaruh kontroler PI muncul pada frekuensi rendah.



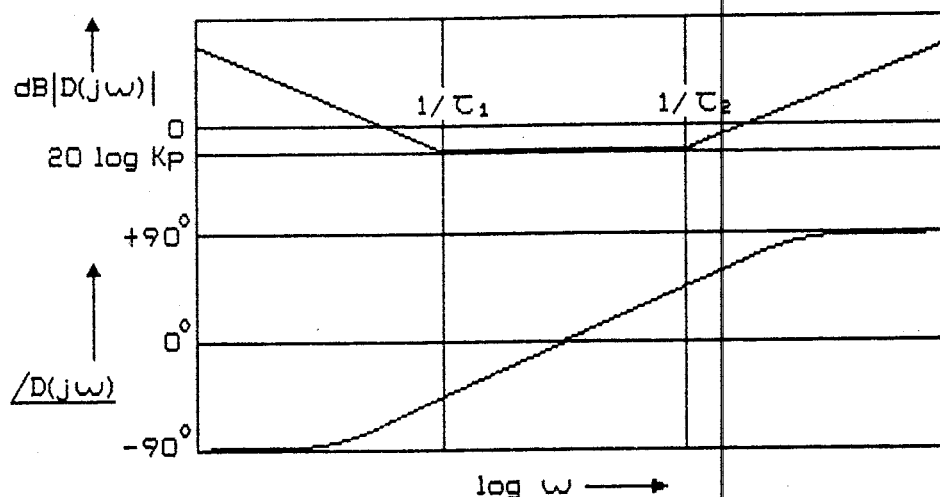
Gambar 2-4<sup>3)</sup>

#### RESPON FREKUENSI KONTROLER PD

Kontroler PID dengan fungsi alih persamaan (2.1) adalah gabungan dari kedua kompensator di atas. Respon frekuensi kontroler PID seperti gambar 2-5.

Prinsip kontrol PID analog dapat dikembangkan ke bentuk digital. Dalam bentuk digital, kontrol proporsional diimplementasikan dengan konstanta proporsional  $K_p$  seperti dalam bentuk analog. Untuk mengimplementasikan integrasi dan turunan secara digital, dapat dilakukan dengan pendekatan.

3) Ibid., p.138

Gambar 2-5<sup>4)</sup>

## RESPON FREKUENSI KONTROLER PID

Pendekatan bentuk integral untuk waktu diskrit dapat dilakukan sebagai berikut. Misalkan  $u(t)$  adalah integral dari  $e(t)$ . Maka harga integral pada  $t=(k+1)T$  sama dengan harga pada  $kT$  ditambah luas daerah dari  $kT$  sampai  $(k+1)T$ .

$$u[(k+1)T] = u(kT) + \int_{kT}^{(k+1)T} e(\sigma) d\sigma$$

Dengan menggunakan hukum trapezoidal, luas daerah di bawah kurva  $e(t)$  dari  $t=kT$  hingga  $t=(k+1)T$  dapat didekati sebagai berikut:

$$\frac{e[(k+1)T] + e(kT)}{2} \times T$$

---

4) Ibid., p.139

Jadi:

$$u[(k+1)T] = u(kT) + \frac{T}{2} \{e[(k+1)T] + e(kT)\} \quad (2.2)$$

Dengan transformasi-z, kita dapatkan:

$$zU(z) = U(z) + \frac{T}{2} [zE(z) + E(z)]$$

Maka:

$$\frac{U(z)}{E(z)} = \frac{T}{2} \left[ \frac{z + 1}{z - 1} \right] \quad (2.3)$$

Jadi Persamaan (2.3) adalah fungsi alih untuk integrator diskrit.

Selanjutnya mengenai pendekatan bentuk turunan untuk waktu diskrit. Jika turunan dari  $e(t)$  pada  $t=kT$  adalah  $u(kT)$ , maka:

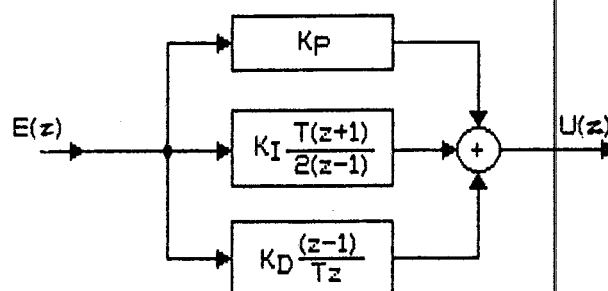
$$u(kT) \approx \frac{e(kT) - e[(k-1)T]}{T}$$

Jadi:

$$\frac{U(z)}{E(z)} = \frac{(z - 1)}{Tz} \quad (2.4)$$

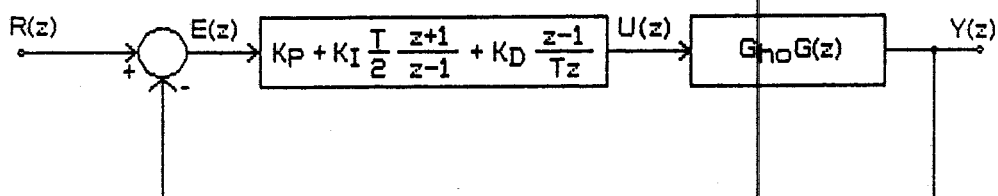
Sebenarnya banyak cara yang dapat dilakukan untuk pendekatan waktu diskrit pada integral dan turunan. Namun dalam pembahasan ini akan digunakan persamaan (2.3) untuk integral dan (2.4) untuk turunan. Diagram blok Kontroler PID digital ditunjukkan pada gambar 2-6. Metoda konversi dari  $U(z)/E(z) = D(z)$  menjadi algoritma numerik akan dibahas pada bagian berikutnya.



Gambar 2-6<sup>5)</sup>

## KONTROLER PID DIGITAL

Desain Kontroler PID digital pada prinsipnya melibatkan pembatasan harga-harga  $K_p$ ,  $K_I$  dan  $K_D$  sehingga sistem yang dikontrol bekerja sesuai dengan yang diharapkan. Berikut ini adalah prosedur desain yang menghasilkan phase margin tertentu tetapi tidak mengatur gain margin/bandwidth.

Gambar 2-7<sup>6)</sup>

## SISTEM KONTROL DIGITAL dengan KONTROLER PID

Fungsi alih Kontroler PID dalam bidang- $w'$  didapat sebagai berikut:

---

5) Ibid., p.140

6) Loc. cit.

$$\begin{aligned}
 D(w') &= K_P + K_I \frac{T}{2} \left[ \frac{z+1}{z-1} \right] \bigg|_{z=\frac{1+w'T/2}{1-w'T/2}} + K_D \frac{z-1}{Tz} \bigg|_{z=\frac{1+w'T/2}{1-w'T/2}} \\
 &= K_P + \frac{K_I}{w'} + K_D \left[ \frac{w'}{1+w'T/2} \right] \quad (2.5)
 \end{aligned}$$

Maka:

$$\begin{aligned}
 D(jv') &= K_P - j \frac{K_I}{v'} + \frac{K_D j v'}{1+jv'T/2} \\
 &= \left[ K_P + \frac{K_D v'^2 (2/T)}{(2/T)^2 + v'^2} \right] + j \left[ \frac{K_D v' (2/T)^2}{(2/T)^2 + v'^2} - \frac{K_I}{v'} \right] \\
 &= |D(jv')| \angle \theta \quad (2.6)
 \end{aligned}$$

Permasalahan desain sekarang adalah memilih  $D(w')$  sedemikian hingga:

$$D(jv') G_{ho} G(jv') = 1 \angle 180^\circ + \phi_s \quad (2.7)$$

$\phi_s$  = phase margin tertentu pada titik potong antara gain dan frekuensi  $v_c'$  yang dipilih.

Dari persamaan (2.5):

$$\begin{aligned}
 &\left[ K_P + \frac{K_D v_c'^2 (2/T)}{(2/T)^2 + v_c'^2} \right] + j \left[ \frac{K_D v_c' (2/T)^2}{(2/T)^2 + v_c'^2} - \frac{K_I}{v_c'} \right] \\
 &= |D(jv_c')| (\cos \theta + j \sin \theta) \quad (2.8)
 \end{aligned}$$

$$\text{dimana} \quad \theta = 180^\circ + \phi_s - \angle G_{ho} G(jv_c') \quad (2.9)$$

Dari persamaan (2.7) dan (2.8),

$$K_P + \frac{K_D v_c'^2 (2/T)}{(2/T)^2 + v_c'^2} = \frac{\cos \theta}{|G_{ho} G(jv_c')|} \quad (2.10a)$$

dan

$$\frac{K_D v_c' (2/T)^2}{(2/T)^2 + v_c'^2} - \frac{K_I}{v_c'} = \frac{\sin \theta}{|G_{ho} G(jv_c')|} \quad (2.10b)$$

Persamaan (2.9) dan (2.10) adalah persamaan desain. Persamaan-persamaan ini juga berlaku untuk desain kontroler PI dan PD dengan penguatan ( $K_D$  atau  $K_I$ ) diberi harga nol.

Untuk Kontroler PI,

$$K_P = \frac{\cos\theta}{|G_{ho}G(jv_c')|} \quad (2.11a)$$

$$-\frac{K_I}{v_c'} = \frac{\sin\theta}{|G_{ho}G(jv_c')|} \quad (2.11b)$$

Untuk suatu plant  $G_{ho}G(w')$  tertentu, pemilihan  $v_c'$  dan  $\phi_s$  akan menentukan harga  $K_P$  dan  $K_I$ .

Untuk Kontroler PD, persamaan (2.10) didapat:

$$K_D = \left[ \frac{\sin\theta}{|G_{ho}G(jv_c')|} \right] \left[ \frac{(2/T)^2 + v_c'^2}{v_c'(2/T)^2} \right]$$

$$K_P = \frac{\cos\theta}{|G_{ho}G(jv_c')|} - \frac{K_D v_c'^2 (2/T)}{(2/T)^2 + v_c'^2}$$

Kedua konstanta  $K_P$  dan  $K_D$  yang tidak diketahui bisa diperoleh untuk plant  $G_{ho}G(w')$  tertentu, dengan memilih harga  $v_c'$  dan  $\phi_s$  tertentu.

Pada Kontroler PID, tidak ada prosedur yang sederhana untuk menyelesaikan persamaan (2.10) untuk  $K_P, K_I$  dan  $K_D$ . Akan tetapi, jika  $v_c' \ll 2/T$ , persamaan (2.5) dapat direduksi menjadi:

$$\begin{aligned} D(w') &= K_P + \frac{K_I}{w'} + K_D w' \\ &= \underbrace{\left[ K_P' + \frac{K_I}{w'} \right]}_{\text{Bagian PI}} \underbrace{(K_P'' + K_D' w')}_{\text{Bagian PD}} \quad (2.12) \end{aligned}$$

Dengan demikian teknik desain pada Kontroler PID dapat dilakukan dengan mengikuti teknik desain untuk Kontroler PI dan Kontroler PD.

### II.2.2 Realisasi Fungsi Alih Pulsa

Pada bagian sebelumnya telah dibahas mengenai prosedur-prosedur yang menghasilkan algoritma kontrol dalam bentuk fungsi alih pulsa. Jika fungsi alih ini disusun ke dalam rumus state diagram atau state space, akan dapat secara langsung diprogram pada komputer. Pada bagian ini akan dibahas mengenai realisasi fungsi alih pulsa ke dalam state model.

Fungsi alih rasional  $H(z)$  dapat direalisasi dengan suatu state model time-invariant dimensi linier ter-hingga bila dan hanya bila  $H(z)$  adalah suatu pecahan yang bisa diselesaikan. Derajat polinomial pembilang adalah kurang dari atau sama dengan polinomial penyebut.

Jika fungsi alih pulsa  $H_1(z)$  adalah:

$$H_1(z) = \frac{b_0 z^n + b_1 z^{n-1} + \dots + b_n}{z^n + a_1 z^{n-1} + \dots + a_n} = \frac{Y_1(z)}{U(z)} \quad (2.13)$$

Bentuk ini dapat dinyatakan dalam state model dengan bentuk:

$$x(k) = Fx(k-1) + gu(k-1) \quad (2.14a)$$

$$y_1(k) = cx(k) + du(k) \quad (2.14b)$$

dimana  $F$  adalah matriks konstan  $n \times n$ ,  $g$  adalah vektor konstan  $n \times 1$ ,  $c$  adalah vektor konstan  $1 \times n$  dan  $d$  adalah konstanta skalar.

Dari persamaan (2.13) jelas bahwa jika harga polinomial pembilang lebih besar daripada polinomial penyebut, maka perluasan  $H_1(z)$  ke dalam deretan  $z^{-1}$  akan mengandung harga positif dalam  $z$  yang menunjukkan 'prediksi', yaitu output mendahului input.

Dengan menggunakan pembagian yang panjang,  $H_1(z)$  pada persamaan (2.13) dapat ditulis sebagai:

$$H_1(z) = \frac{\beta_1 z^{n-1} + \beta_2 z^{n-2} + \dots + \beta_n}{z^n + \alpha_1 z^{n-1} + \dots + \alpha_n} + b_0 \quad (2.15a)$$

$$= \frac{N(z)}{\Delta(z)} + b_0 \quad (2.15b)$$

Dari persamaan (2.14) dan (2.15) dengan mudah diketahui bahwa  $d = b_0$ . Dengan bagian transmisi langsung dari state model yang telah didapat, maka untuk selanjutnya fungsi rasional di atas menjadi:

$$H(z) = \frac{\beta_1 z^{n-1} + \beta_2 z^{n-2} + \dots + \beta_n}{z^n + \alpha_1 z^{n-1} + \dots + \alpha_n} = \frac{Y(z)}{U(z)} \quad (2.16)$$

sehingga fungsi alih ini dapat dinyatakan ke dalam state model dengan bentuk:

$$x(k) = Fx(k-1) + gu(k-1) \quad (2.17a)$$

$$y(k) = cx(k) \quad (2.17b)$$

Pada persamaan (2.16) derajat polinomial pembilang adalah  $(n-1)$ . Jika derajatnya kurang dari  $(n-1)$ , maka hasil yang didapat bisa langsung digunakan dengan memberikan harga koefisien  $\beta_1$  sama dengan nol.

Di sini juga akan digunakan rumus penguatan Mason untuk grafik aliran sinyal yang menghubungkan fungsi

alih dengan grafik aliran sinyal (signal flow graph).

Rumus ini menyatakan,

$$H(z) = \frac{1}{\Delta} \sum_k P_k \Delta_k \quad (2.18)$$

dimana

$P_k$  = penguatan lintasan maju ke-k

$\Delta$  = determinan grafik

= 1- (jumlah semua penguatan loop yang berbeda) +  
(jumlah hasil kali penguatan dari semua kombinasi  
yang mungkin dari dua loop yang tidak bersentuhan)  
- (jumlah hasil kali penguatan dari semua  
kombinasi yang mungkin dari tiga loop yang tidak  
bersentuhan) + ...

$\Delta_k$  = kofaktor dari determinan lintasan maju ke-k  
dengan menghilangkan loop-loop yang menyentuh  
lintasan maju ke-k.

Misalkan pada persamaan(2.16) diambil  $n = 3$ , maka:

$$\frac{Y(z)}{U(z)} = H(z) = \frac{\beta_1 z^2 + \beta_2 z + \beta_3}{z^3 + \alpha_1 z^2 + \alpha_2 z + \alpha_3} \quad (2.19)$$

Ini merupakan fungsi alih order-tiga, sehingga terdapat  
tiga state variable  $x_1$ ,  $x_2$  dan  $x_3$ . State diagram akan  
mempunyai tiga unit penunda  $z^{-1}$ .

Persamaan (2.19) juga dapat disusun sebagai:

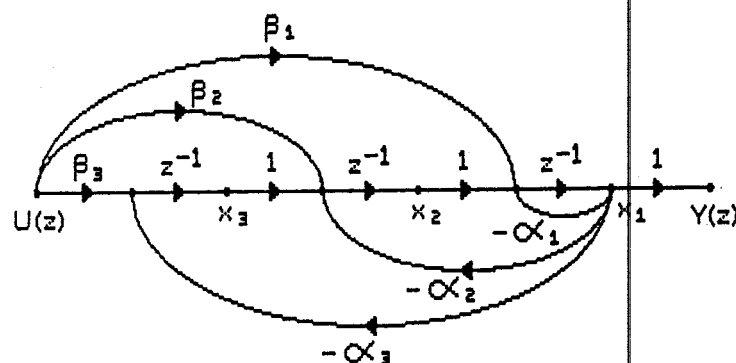
$$H(z) = \frac{\beta_1 z^{-1} + \beta_2 z^{-2} + \beta_3 z^{-3}}{1 - (-\alpha_1 z^{-1} - \alpha_2 z^{-2} - \alpha_3 z^{-3})} \quad (2.20)$$

Dengan membandingkan persamaan (2.20) dan (2.18),  
diketahui bahwa grafik aliran sinyal pada persamaan  
(2.20) terdiri dari:

- (i) tiga loop umpan balik (saling bersentuhan) dengan gain  $-\alpha_1 z^{-1}$ ,  $-\alpha_2 z^{-2}$  dan  $-\alpha_3 z^{-3}$
- (ii) tiga lintasan maju yang menyentuh loop dan mempunyai penguatan  $\beta_1 z^{-1}$ ,  $\beta_2 z^{-2}$  dan  $\beta_3 z^{-3}$ .

Konfigurasi grafik aliran sinyal (state diagram) yang memenuhi syarat untuk ini ditunjukkan pada gambar 2-8. Dengan mengambil output dari unit penunda sebagai state variable, akan didapat

$$\begin{aligned}x_1(k) &= -\alpha_1 x_1(k-1) + x_2(k-1) + \beta_1 u(k-1) \\x_2(k) &= -\alpha_2 x_1(k-1) + x_2(k-1) + \beta_2 u(k-1) \\x_3(k) &= -\alpha_3 x_1(k-1) + \beta_3 u(k-1) \\y(k) &= x_1(k)\end{aligned}\tag{2.21}$$



Gambar 2-8<sup>7)</sup>

STATE DIAGRAM UNTUK PERSAMAAN (2.20)

### II.2.3 Realisasi Fungsi Alih Kontroler FID Digital

Berdasarkan teori yang telah dibahas pada bagian II.2.2 dan dengan memperhatikan diagram blok Kontroler

7) Ibid., p.197

PID pada gambar 2-6, maka fungsi alih Kontroler PID Digital dapat diselesaikan dalam bentuk algoritma numerik.

Dari diagram blok pada gambar 2-6, fungsi alih Kontroler PID digital adalah:

$$\frac{U(z)}{E(z)} = K_P + K_I \frac{T}{2} \left[ \frac{z+1}{z-1} \right] + K_D \frac{1}{T} \left[ \frac{z-1}{z} \right]$$

atau

$$\begin{aligned} \frac{U(z)}{E(z)} &= K_P + K_I \frac{T}{2} \left[ \frac{1+z^{-1}}{1-z^{-1}} \right] + K_D \frac{1}{T} (1-z^{-1}) \\ &= \frac{K_P(1-z^{-1}) + K_I(T/2)(1+z^{-1}) + K_D/T(1-z^{-1})^2}{1-z^{-1}} \\ &= \frac{[K_P+K_I(T/2)+K_D/T] + [K_I(T/2)-K_P-2(K_D/T)]z^{-1} + (K_D/T)z^{-2}}{1-z^{-1}} \end{aligned}$$

bila  $K_1 = K_P+K_I(T/2)+K_D/T,$   
 $K_2 = K_I(T/2)-K_P-2(K_D/T)$  dan  
 $K_3 = K_D/T$

Maka:

$$\frac{U(z)}{E(z)} = \frac{K_1 + K_2 z^{-1} + K_3 z^{-2}}{1 - z^{-1}} \quad (2.22)$$

Persamaan (2.22) merupakan fungsi alih order dua dan mempunyai state variable  $x_1$  dan  $x_2$ . Grafik aliran sinyal seperti gambar 2-9.

Dari gambar 2-9 diperoleh:

$$(a.) \quad x_1(k) = x_1(k-1) + x_2(k-1) + K_1 e(k)$$

$$(b.) \quad x_2(k) = K_2 e(k) + K_3 e(k-1), \quad \text{atau}$$

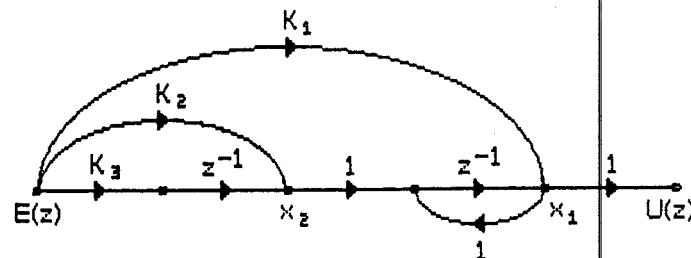
$$x_2(k-1) = K_2 e(k-1) + K_3 e(k-2)$$



(c.)  $u(k) = x_1(k)$  dan  $u(k-1) = x_1(k-1)$

Substitusi (b) ke (a) diperoleh:

$$x_1(k) = x_1(k-1) + K_2 e(k-1) + K_3 e(k-2) + K_1 e(k) \quad (2.23)$$



Gambar 2-9

STATE DIAGRAM UNTUK PERSAMAAN (2.22)

Selanjutnya dengan substitusi (c) ke persamaan (2.23) didapat:

$$u(k) = u(k-1) + K_1 e(k) + K_2 e(k-1) + K_3 e(k-2) \quad (2.24)$$

Persamaan (2.24) merupakan algoritma yang akan diimplementasikan dalam prosesor.

### II.3 MOTOR ARUS SEARAH (DC)

Motor DC ialah suatu mesin yang berfungsi mengubah tenaga listrik arus searah menjadi tenaga gerak atau tenaga mekanik yang berupa putaran pada rotor.

Secara konstruksi tidak ada perbedaan antara motor DC dan generator DC. Pada prinsipnya motor DC bisa dipakai sebagai generator DC, sebaliknya generator DC bisa dipakai sebagai motor DC.

### II.3.1 Prinsip Dasar Motor DC

Prinsip dasar motor arus searah adalah bila sebuah kawat berarus diletakkan antara kutub magnet (U-S) maka pada kawat itu akan bekerja suatu gaya yang menggerakkan kawat itu.

Arah gerakan kawat sesuai dengan "kaidah tangan kiri" yang berbunyi sebagai berikut :

"Apabila tangan kiri terbuka diletakkan di antara kutub Utara (U) dan Selatan (S) sehingga garis-garis gaya yang keluar dari kutub Utara menembus telapak tangan kiri dan arus di dalam kawat mengalir searah dengan arah keempat jari, maka kawat itu akan mendapat gaya yang arahnya sesuai dengan arah ibu jari".

Besarnya gaya yang ditimbulkan :

$$F = B i l \text{ Newton} \quad (2.25)$$

dimana  $B$  = kepadatan fluks magnet (satuan Weber).

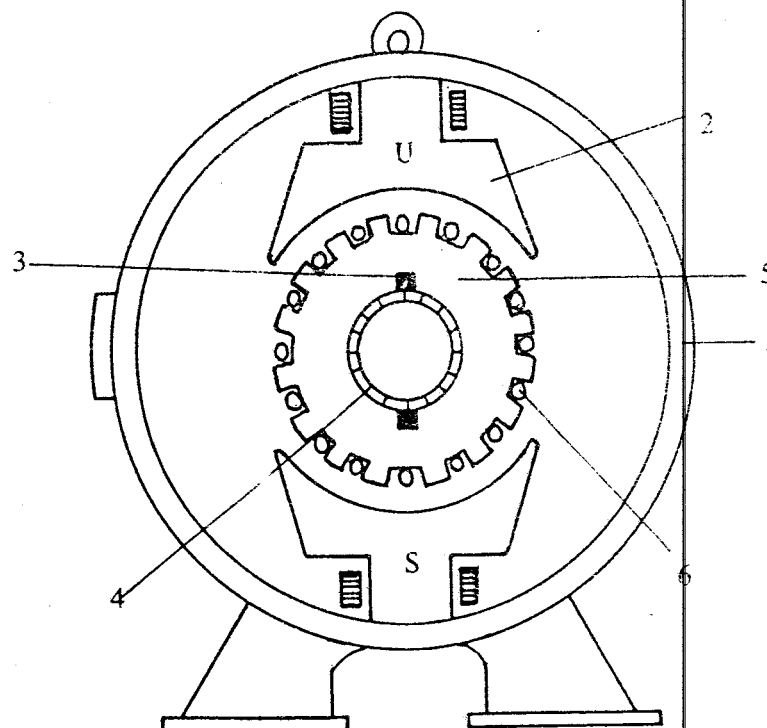
$i$  = arus listrik yang mengalir (satuan Ampere).

$l$  = panjang kawat penghantar (satuan meter).

### II.3.2 Konstruksi Motor DC

Motor arus searah mempunyai bagian yang diam (stator) dan bagian yang berputar (rotor). Bagian stator terdiri dari badan motor (body) yang memiliki lempeng-lempeng kutub magnet yang melekat padanya. Untuk motor kecil, lempengan-lempengan tersebut adalah magnet permanen. Sedang pada motor yang besar berupa elektromagnetik. Umumnya lempengan-lempengan magnet

terbuat dari lempeng-lempeng baja magnetik derajat tinggi. Kumparan yang dililitkan pada lempeng-lempeng magnet ini disebut *kumparan medan*.



Gambar 2-10<sup>8)</sup>

#### BAGIAN-BAGIAN MOTOR DC

1. Badan motor, 2. Inti kutub magnet dan lilitan penguat magnet, 3. Sikat-sikat,  
4. Komutator, 5. Jangkar, 6. Lilitan jangkar.

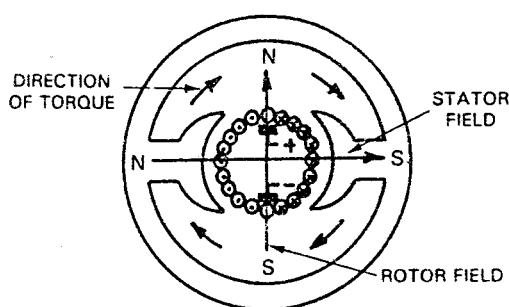
Rotor terdiri dari jangkar yang intinya terbuat dari lempengan-lempengan yang ditakik. Susunan lem-

8) Sumarto, Drs., 1984, Mesin Arus Searah, hal.21

pengan-lempengan membentuk celah-celah. Konduktor kumparan jangkar dimasukkan pada celah-celah tersebut. Ujung dari tiap-tiap kumparan dihubungkan pada satu segmen *Komutator*. Tiap segmen merupakan pertemuan dua ujung kumparan yang terhubung ke segmen itu. Gambar 2-10 menunjukkan bagian-bagian motor DC.

### II.3.3 Torsi Motor

Bila kumparan medan diberi tegangan, maka akan terbentuk kutub-kutub magnet utara dan selatan pada lempeng-lempeng magnet. Pada kumparan jangkar juga diberi tegangan melalui komutator dan sikat-sikat. Hal ini menyebabkan terbentuknya medan magnet di sekeliling masing-masing konduktor. Maka akan menghasilkan torsi seperti tampak pada gambar 2-11.



Gambar 2-11<sup>9)</sup>

TORSI PUTAR PADA MOTOR DC

9) Joe Kaiser., 1982, Electrical Power, p. 209



tansi  $R_a$ . Tegangan jatuh itu sebesar  $I_a R_a$ . Jadi :

$$V = E_b + I_a R_a \quad (2.29)$$

Dari bentuk ini dapat dituliskan :

$$V I_a = E_b I_a + I_a^2 R_a \quad (2.30)$$

dengan

$V I_a$  = daya listrik yang masuk ke jangkar

$E_b I_a$  = daya mekanik yang setara dengan daya listrik yang timbul dalam jangkar =  $P_m$

$I_a^2 R_a$  = rugi tembaga dalam jangkar

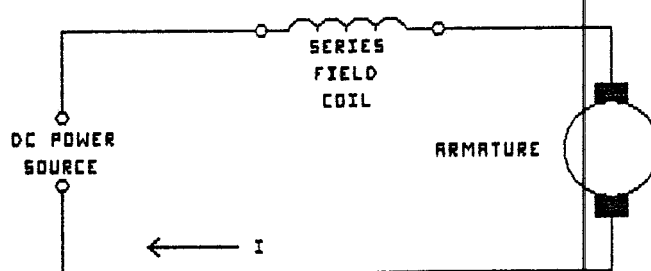
### II.3.5 Jenis-jenis Motor DC

Berdasarkan sumber arus penguat magnetnya, motor DC dapat dibedakan atas :

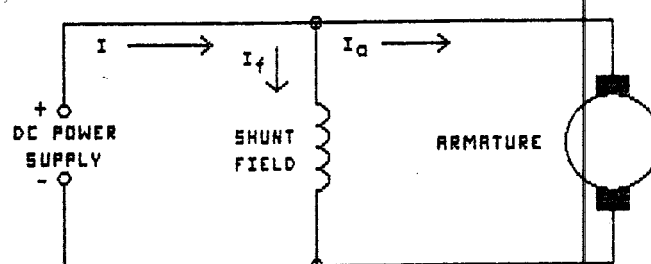
- Motor DC penguat terpisah :  
     arus penguat medan magnet diperoleh dari sumber DC diluar motor.
- Motor DC penguat sendiri :  
     arus penguat medan magnet berasal dari motor itu sendiri.

Berdasarkan hubungan lilitan penguat magnet terhadap lilitan jangkar motor DC dibedakan atas :

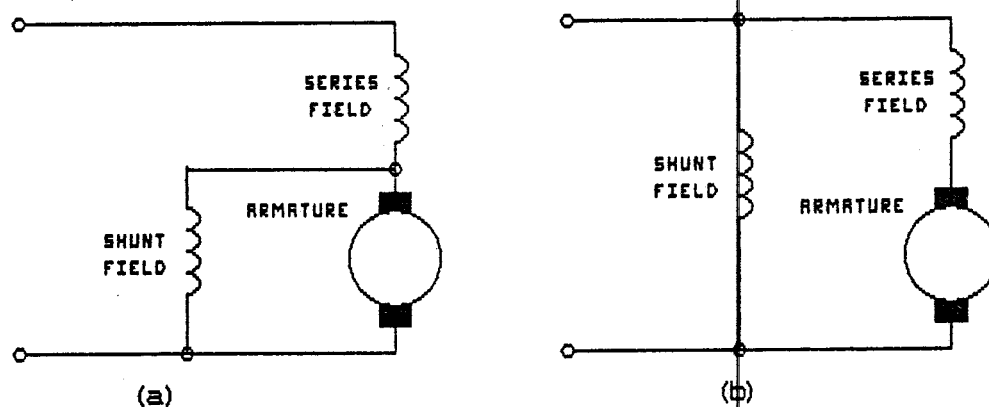
- Motor DC Seri (gambar 2-12)
- Motor DC Shunt (gambar 2-13)
- Motor Kompon yang dibedakan lagi menjadi motor kompon pendek (gambar 2-14(a)) dan motor kompon panjang (gambar 2-14(b)).

Gambar 2-12<sup>10)</sup>

RANGKAIAN PENGGANTI UNTUK MOTOR DC SERI

Gambar 2-13<sup>11)</sup>

RANGKAIAN PENGGANTI UNTUK MOTOR DC SHUNT

Gambar 2-14<sup>12)</sup>

RANGKAIAN PENGGANTI UNTUK MOTOR DC KOMPON

10) *Ibid.*, p. 21011) *Ibid.*, p. 21212) *Ibid.*, p. 215

### II.3.6 Motor DC Sebagai Motor Servo

Dalam sistem servomekanisme yang menggunakan motor servo DC, konstruksi dasar motor biasanya sama dengan motor DC shunt yang umum. Perbedaannya yaitu terletak pada cara menghubungkan kumparan jangkar dan kumparan medan terhadap sumber tegangan. Pada motor DC shunt kedua kumparan dihubungkan secara paralel dan diberi tegangan dari sumber yang sama, tetapi pada motor servo kumparan-kumparannya diberi tegangan dari sumber yang terpisah. Jadi motor servo sebenarnya menggunakan penguat terpisah. Hal ini ditunjukkan pada gambar 2-15.

Untuk membalik arah putaran motor dapat dilakukan dengan dua cara yaitu dengan membalik polaritas magnet pada kumparan medan atau membalik arah arus jangkar. Akan tetapi bukan kedua-duanya, karena yang demikian tidak akan mengubah arah putaran motor.

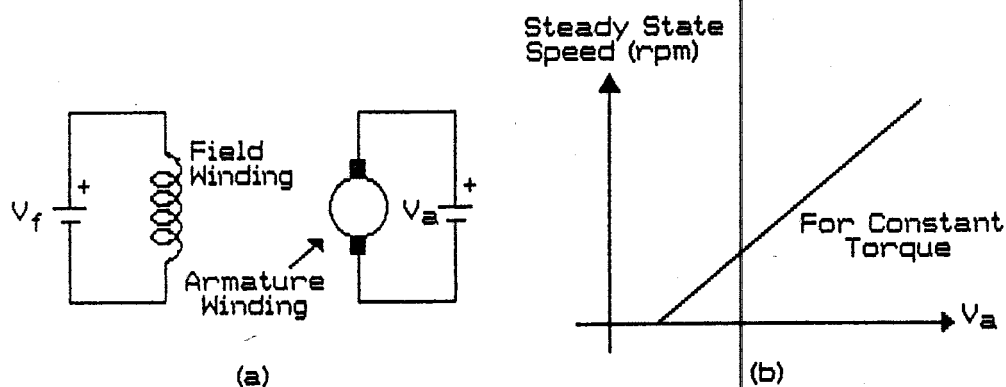
Di sini tidak akan dibahas secara detil mengenai pengaturan kecepatan motor, tapi cukup diketahui bahwa kecepatan pada keadaan tunak dapat dikontrol dengan mengubah  $V_f$  atau  $V_a$  seperti ditunjukkan pada gambar 2-15(a). Dalam sistem servo modern, umumnya pengaturan dilakukan terhadap  $V_a$  (tegangan pada jangkar).

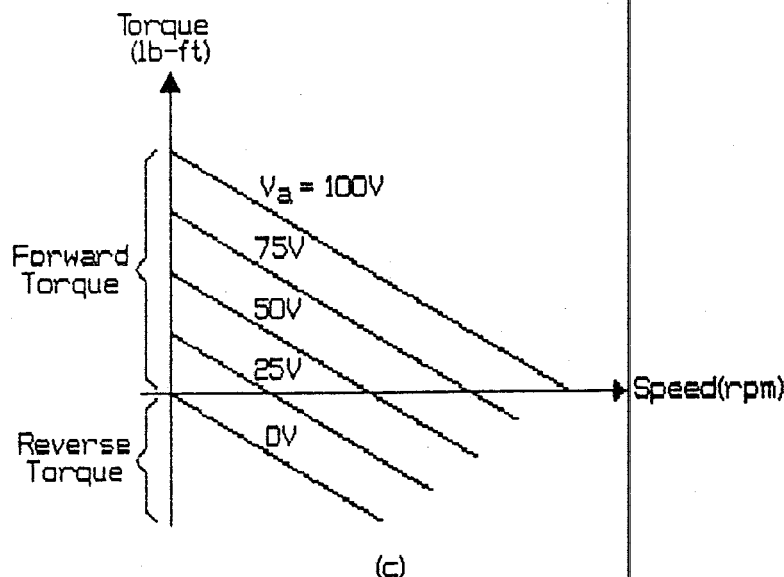
Hubungan antara kecepatan keadaan tunak (steady state speed) dan tegangan jangkar ditunjukkan pada gambar 2-15(b) untuk torsi konstan. Pada gambar tersebut terlihat bahwa hubungan keduanya mendekati linier. Akan tetapi pada sistem servo kenyataannya torsi motor tidaklah konstan. Ia berubah saat obyek yang dikontrol



mendekati posisi yang dikehendaki dan tegangan motor berkurang. Kurva torsi-kecepatan pada tegangan jangkar yang bervariasi ditunjukkan pada gambar 2-15(c).

Kurva tersebut menunjukkan bahwa motor servo DC menghasilkan torsi yang lebih besar pada kecepatan rendah daripada kecepatan tinggi untuk tegangan jangkar  $V_a$  yang diberikan. Hal ini memungkinkan motor memberikan percepatan pada beban (obyek yang diatur posisinya) dari keadaan diam dalam waktu cepat. Selain itu kurva torsi-kecepatan menunjukkan bahwa saat obyek yang diatur mendekati posisi yang diinginkan dan  $V_a$  berkurang, motor DC mampu memberikan torsi balik untuk memperlambat gerak beban jika kecepatannya terlalu tinggi. Hal ini bisa terjadi karena pada kondisi ini kumparan jangkar motor mulai bekerja seperti generator. Arus pada konduktor-konduktor jangkar berbalik arah dan arus balik ini menimbulkan torsi balik.



Gambar 2-15<sup>13)</sup>

## MOTOR SERVO DC

(a)Skema motor servo DC. (b)Grafik kecepatan vs tegangan jangkar

untuk torsi poros konstan. (c)Kurva torsi vs kecepatan untuk berbagai tegangan jangkar.

Motor DC Shunt adalah cocok sebagai motor servo. Ia memiliki torsi besar pada kecepatan rendah dan memiliki suatu *efek redaman* yang tetap, sehingga pada kecepatan pendekatan (menuju posisi yang diinginkan) yang tinggi mengakibatkan kecenderungan perlambatan secara otomatis.

## II.4 KONVERTER DATA

Instrumen-instrumen analog dan digital merupakan instrumen yang umum dipakai dalam aplikasi elektronika. Masing-masing dapat bekerja secara independen atau bekerja bersama-sama dalam suatu sistem.

13) Maloney, Timothy J., 1986, Industrial Solid-State Electronics, p.382

Dalam dunia analog, sinyal bekerja secara kontinyu yang bervariasi antara jangkauan batas atas dan batas bawah. Tetapi sinyal-sinyal pada rangkaian digital bekerja secara diskrit yang berupa sederetan keadaan yang dinyatakan dengan logika "0" dan "1".

Konverter data adalah suatu rangkaian atau peralatan yang mengubah sinyal dari bentuk asalnya ke besaran lain yang harganya sebanding. Misalnya digital to analog converter (DAC) mengkonversi word digital yang terdiri dari bit-bit bilangan tertentu menjadi tegangan atau arus yang mewakili harga bilangan biner dari word digital tersebut. Misalnya DAC 8-bit menghasilkan sinyal output 0 volt bila pada inputnya diberikan harga biner  $00000000_2$  dan menghasilkan 2,56 volt bila diberikan input  $11111111_2$ . Jadi untuk setiap harga biner yang diberikan, maka akan dihasilkan harga output yang sebanding.

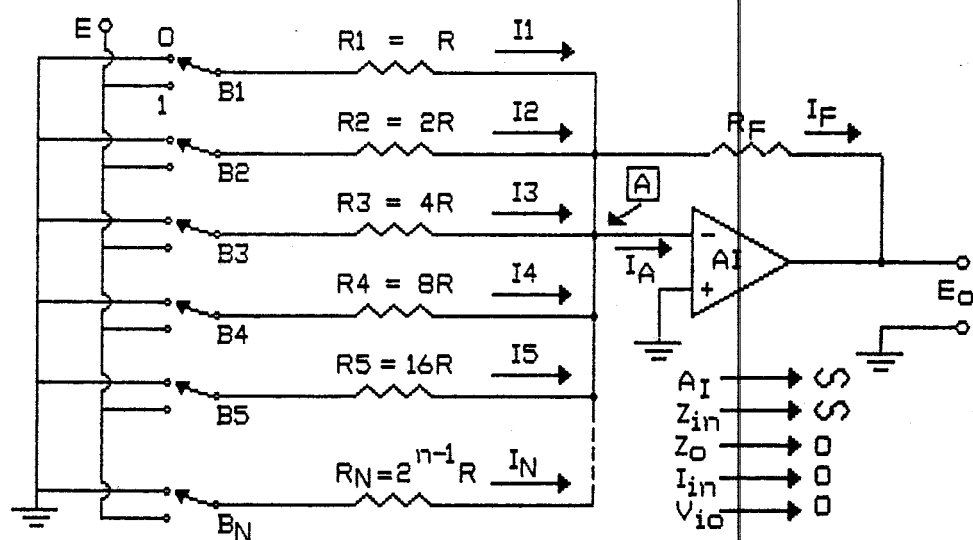
Analog to digital converter (ADC) adalah kebalikan dari DAC yaitu mengkonversi sinyal analog (tegangan atau arus) menjadi word biner yang sebanding.

Konverter data terutama banyak digunakan pada aplikasi interface suatu transduser dengan peralatan digital atau input komputer; dan interface dari output suatu instrumen digital dengan perangkat analog seperti motor, alat ukur analog dan lain-lain.

#### II.4.1 Rangkaian Dasar DAC

Rangkaian dasar DAC yang paling mudah adalah

*binary weighed resistor DAC* yang ditunjukkan pada gambar 2-16. Resistor-resistor itu disebut *binary weighed* karena harga-harganya mempunyai perbandingan dua. Atau lebih jelasnya, bila harga resistor terendah adalah  $R$ , maka resistor urutan berikutnya berharga  $2R$ ,  $4R$ ,  $8R$ ,  $16R$  dan seterusnya hingga resistor ke- $n$  yang berharga  $2^{(n-1)}R$ . Di sini tampak bahwa semakin besar



Gambar 2-16<sup>14)</sup>

#### RANGKAIAN *BINARY WEIGHED RESISTOR DAC*

jumlah bit maka perbedaan nilai resistor akan semakin besar pula, 2 kali lipat setiap penambahan 1 bit.

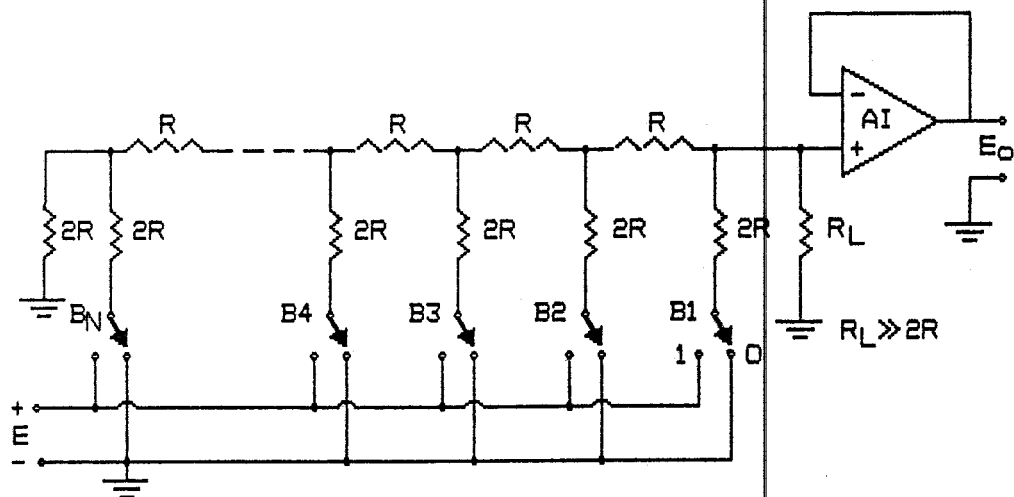
Tegangan keluaran DAC pada gambar 2-16 adalah :

$$E_o = -R_f \cdot E \left( D_1 \frac{1}{R} + D_2 \frac{1}{2R} + \dots + D_n \frac{1}{2^{(n-1)}R} \right) \quad (2.31)$$

14) Carr. Joseph J., 1979, Elements of Electronic Instrumentation and Measurement, p.316

dengan  $D_1$  s/d  $D_n$  menyatakan keadaan saklar, membuka berarti 0 dan menutup berarti 1.  $D_1$  merupakan MSB dan  $D_n$  adalah LSB.

Jenis rangkaian DAC yang lain adalah *R-2R resistor ladder DAC* seperti ditunjukkan pada gambar 2-17.



Gambar 2-17<sup>15)</sup>

#### R-2R RESISTOR LADDER DAC

Rangkaian ini banyak digunakan pada DAC yang dibuat untuk keperluan komersial karena mempunyai kelebihan dibanding binary weighed resistor DAC dalam hal pemakaian resistor yang hanya 2 macam, R dan 2R. Tapi konsekuensinya yaitu harus menggunakan jumlah resistor yang dua kali lebih banyak.

Tegangan keluaran DAC pada gambar 2-17 adalah :

$$E_o = E \left( \frac{D_1}{2} + \frac{D_2}{2^2} + \dots + \frac{D_{n-1}}{2^{n-1}} + \frac{D_n}{2^n} \right) \quad (2.32)$$

15) Ibid., p.318

Harga  $R_L \gg 2R$  agar efek pembagi tegangan antara tangga (ladder) dan  $R_L$  dapat diabaikan.

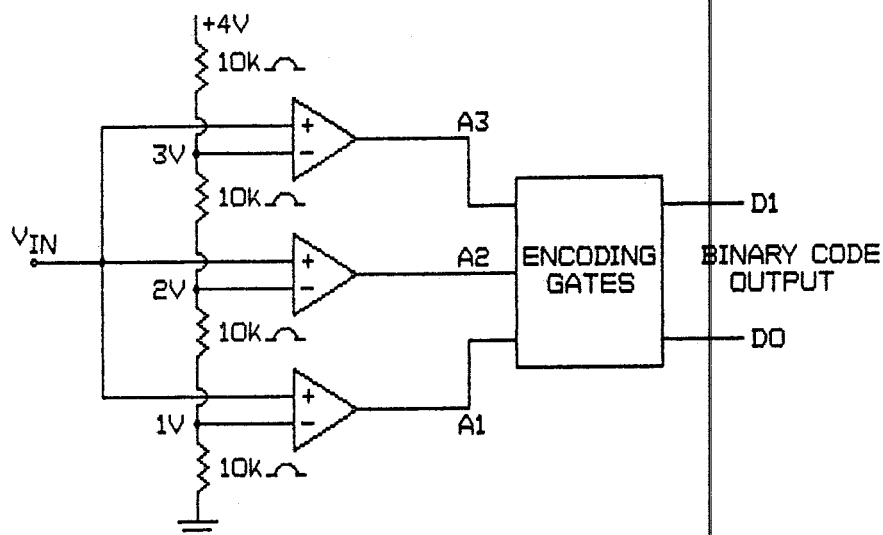
#### II.4.2 Rangkaian Dasar ADC

ADC adalah rangkaian yang dapat mengubah besaran analog menjadi bentuk digital. Dengan pengubahan ini maka sinyal analog dapat diproses secara digital, misalnya dengan komputer. Ada beberapa macam rangkaian ADC yaitu :

##### - Parallel Comparator / Flash ADC

Pada gambar 2-18 ditunjukkan rangkaian parallel comparator ADC. Pada masing-masing input membalik (inverting input) diberikan tegangan referensi yang didapat dari pembagi tegangan. Sinyal yang akan dikonversi diberikan pada input tak membalik (non-inverting input) secara paralel. Jika tegangan yang diberikan pada komparator lebih besar dari tegangan referensi maka keluaran komparator menjadi tinggi. Tapi bila masukannya lebih kecil dari tegangan referensi maka menghasilkan keluaran rendah.

Keunggulan dari Parallel Comparator/Flash ADC ini yaitu memiliki waktu konversi cepat. Tetapi kelemahannya adalah memerlukan jumlah komparator yang banyak. Untuk konverter 2 bit saja seperti gambar 2-18 membutuhkan 3 buah komparator. Jadi untuk menghasilkan konverter dengan resolusi  $N$  bit dibutuhkan  $(2^N - 1)$  komparator. Dengan kata lain untuk konverter 8 bit akan memerlukan 255 komparator.

Gambar 2-18<sup>16)</sup>

## RANGKAIAN PARALLEL COMPARATOR/FLASH ADC

## - Successive Approximation ADC

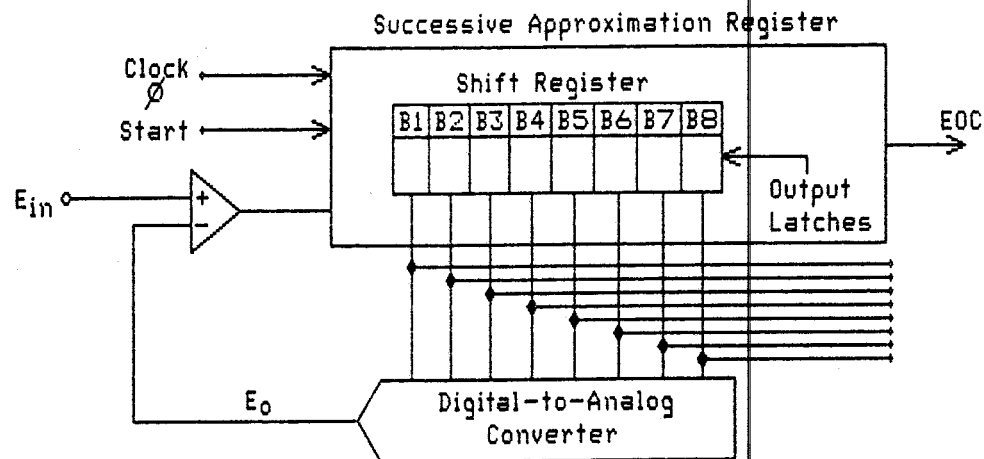
ADC jenis ini banyak dipakai dalam aplikasi interface dengan komputer karena mempunyai waktu konversi cepat disamping dapat dibuat dengan jumlah bit yang banyak/resolusi tinggi. Ciri lain yang dianggap menguntungkan adalah kecepatan konversinya tidak bergantung pada besarnya tegangan sinyal yang dikonversi. Dengan kata lain waktu konversinya konstan.

ADC ini dibentuk dari beberapa komponen yaitu DAC, komparator dan beberapa register serta rangkaian kontrol seperti tampak pada gambar 2-19.

Konversi dilakukan dengan cara membandingkan tegangan input analog terhadap tegangan output yang

16) Douglas V. Hall, 1987, Microprocessors and Interfacing: Programming and Hardware, p.325

dihasilkan oleh ADC. Input DAC diatur oleh register geser dan rangkaian kontrol logika, sedang hasil konversi dapat diambil dari register output.



Gambar 2-19<sup>17)</sup>

#### DIAGRAM BLOK SUCCESSIVE APPROXIMATION ADC

Konversi dimulai dengan memberikan sinyal input 'start conversion'. Adanya sinyal start ini mengakibatkan input DAC dibuat sehingga MSB = '1' dan bit-bit yang lain '0'. Output DAC ( $1/2$  skala penuh) dibandingkan dengan output analog yang akan dikonversi. Bila tegangan output DAC lebih kecil maka bit MSB tetap '1' sedangkan bila output DAC lebih besar maka MSB menjadi '0'. Kemudian bit berikutnya (1 bit lebih rendah dari MSB) dibuat '1' dengan keadaan MSB tetap dipertahankan. Kondisi bit-

17) Joseph J. Carr., *Op. cit.*, p.326



bit pada input DAC yang ada sekarang akan menghasilkan output analog dan dibandingkan lagi dengan tegangan input untuk menentukan apakah bit kedua akan diubah atau tidak. Proses di atas terus berlangsung sampai LSB, sehingga didapat output digital dari masukan DAC yang terakhir.

Pergeseran bit dari MSB hingga LSB dilakukan oleh register geser yang waktunya diatur oleh pulsa clock. Jadi untuk ADC 8 bit memerlukan waktu 8 pulsa clock. Setelah proses konversi selesai maka bit 'output status' akan mengeluarkan sinyal 'End of Conversion' yang menyatakan bahwa output digital ADC dapat diambil sebagai besaran digital yang mewakili tegangan input analog yang dikonversi.

Jenis ADC yang lain dapat disebutkan di sini antara lain :

- Single-slope dan Dual-slope ADC yang menggunakan integrator dan counter untuk mendapatkan besaran digital,
- Tracking ADC yang menggunakan DAC dan counter.

Kedua jenis ADC tersebut juga banyak dipakai pada alat ukur digital yang tidak memerlukan waktu konversi yang cepat, sedangkan untuk aplikasi pemrosesan sinyal digital atau pada kontrol digital yang membutuhkan konversi cepat, kedua jenis ADC ini tidak dapat dipakai mengingat kedua jenis ADC ini kecepatan konversinya kurang.

# BAB III

## MIKROPROSESOR/MIKROKOMPUTER

### TMS32010

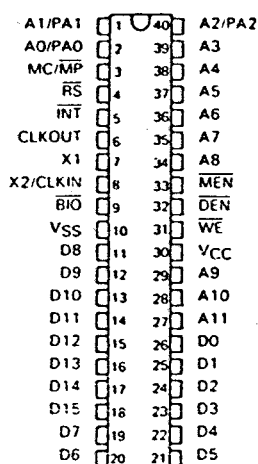
#### III.1 DESKRIPSI UMUM

Prosesor TMS32010 adalah salah satu dari generasi pertama keluarga TMS320, suatu single chip prosesor 16/32 bit yang dirancang khusus untuk aplikasi pengolahan sinyal digital. Prosesor ini memiliki kecepatan eksekusi 6,25 MIPS (million instructions per second) dan mempunyai *hardware multiplier* sehingga mampu mengeksekusi instruksi perkalian 16x16 bit dalam satu instruction cycle.

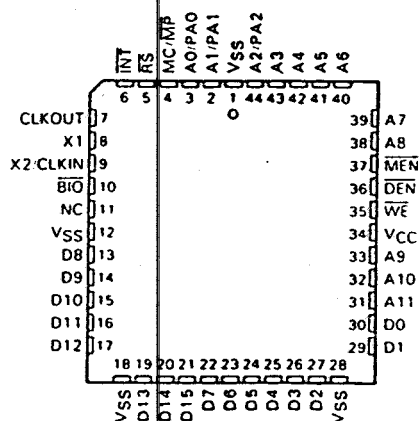
Prosesor sinyal digital TMS32010 seperti keluarga mikroprosesor dan mikrokomputer TMS320 yang lain menggunakan arsitektur Harvard dimana program dan data ditempatkan secara terpisah sehingga overlap dapat terjadi sepenuhnya antara pengambilan instruksi dan eksekusi.

TMS32010 mempunyai *on-chip data memory* atau memori data internal sebanyak 144 word dan dapat mengakses instruksi maksimum sebanyak 4K word dari memori program. Untuk versi mikrokomputer di dalamnya tersedia *on-chip program memory* sebanyak 1,5K word, sedangkan 2,5K word selebihnya bisa ditambahkan di memori program eksternal untuk membentuk memori program total 4K word.

Pada gambar 3-1 ditunjukkan IC prosesor TMS32010 beserta pin-pinnya dalam kemasan DIP dan PLCC.



(a)



(b)

Gambar 3-1<sup>18)</sup>

## IC PROSESOR TMS32010

Adapun deskripsi sinyal-sinyal dari pin-pin prosesor yang ditunjukkan pada gambar 3-1 adalah sebagai berikut:

SINYALDESKRIPSI

- A11-A0/  
PA2-PA0 Bus address memori program A11 (MSB) hingga A0 (LSB) dan address port PA2 (MSB) hingga PA0 (LSB). Address A11-A0 selalu aktif dan tidak pernah berada pada keadaan high-impedance. Selama eksekusi instruksi IN dan OUT, pin-pin A2-A0 berisi address port PA2-PA0 sedangkan address A11-A3 selalu berada pada keadaan low.
- D15-D0 Bus data paralel D15 (MSB) hingga D0 (LSB). Bus data selalu high-impedance kecuali bila  $\overline{WE}$  aktif low.

18) Texas Instruments, 1988, First-Generation TMS320 User's Guide, p. 2-2

$\overline{\text{BIO}}$	Input polling eksternal yang diaktifkan dengan instruksi BIOZ. Jika low, maka device akan branch ke address yang ditentukan oleh instruksi.
$\overline{\text{DEN}}$	Data enable untuk meng-input-kan data pada device. Bila aktif low, $\overline{\text{DEN}}$ menunjukkan bahwa device akan menerima data dari bus data. $\overline{\text{DEN}}$ hanya aktif selama cycle pertama dari instruksi IN. $\overline{\text{MEN}}$ dan $\overline{\text{WE}}$ selalu tidak aktif (high) bila $\overline{\text{DEN}}$ aktif.
$\overline{\text{INT}}$	Input interrupt eksternal. Sinyal interrupt dibangkitkan dengan memberikan negative-going edge pada pin $\overline{\text{INT}}$ . Level aktif low juga bisa digunakan untuk meng-interrupt.
$\text{MC}/\overline{\text{MP}}$	Pin select mode memori yang apabila high maka device berada pada mode mikrokomputer yang memanfaatkan 1,5K word memori program on-chip dari 4K word yang tersedia. Sedangkan 2,5K word selebihnya bisa ditambahkan pada off-chip. Bila sinyal low diberikan pada pin ini maka device berada pada mode mikroprosesor yang seluruh memori programnya adalah eksternal.
$\overline{\text{MEN}}$	Memory enable, yang aktif low pada setiap machine cycle kecuali bila $\overline{\text{WE}}$ dan $\overline{\text{DEN}}$ aktif. $\overline{\text{MEN}}$ merupakan sinyal kontrol yang dibangkitkan oleh device untuk meng-enable fetch

	instruksi dari memori program. $\overline{\text{MEN}}$ akan aktif pada fetch instruksi baik dari memori internal maupun eksternal.
$\overline{\text{RS}}$	Input reset untuk menginisialisasi device. Bila diberikan input aktif low pada pin $\overline{\text{RS}}$ selama minimum lima clock cycle, maka $\overline{\text{DEN}}$ , $\overline{\text{WE}}$ dan $\overline{\text{MEN}}$ akan high dan bus data (D15-D0) menjadi high-impedance. Program counter (PC) dan bus address (A11-A0) secara serentak akan di-clear. Reset juga men-disable interrupt, meng-clear interrupt flag register tapi tidak mengubah overflow mode register. Device dapat berada dalam kondisi reset untuk waktu yang tak terbatas.
$\overline{\text{WE}}$	Write enable untuk meng-output data dari device. Bila aktif low, $\overline{\text{WE}}$ menunjukkan bahwa data akan di-output-kan dari device pada bus data. $\overline{\text{WE}}$ hanya aktif selama cycle pertama dari instruksi OUT dan pada cycle kedua dari instruksi TBLW. Bila $\overline{\text{WE}}$ aktif maka $\overline{\text{MEN}}$ dan $\overline{\text{DEN}}$ selalu tidak aktif (high).
CLKOUT	Output sistem clock (dengan frekuensi 1/4 dari frekuensi kristal/CLKIN). Duty cyclenya lima puluh persen.
V <sub>CC</sub>	Pin supply 5V.
V <sub>SS</sub>	Pin ground.
X1	Pin output kristal untuk osilator internal.

Jika osilator internal tidak digunakan, pin ini harus dibiarkan tak terhubung.

X2/CLKIN Pin input kristal pada osilator internal (X2). Atau pin input clock untuk osilator eksternal (CLKIN).

### III.2 ARSITEKTUR TMS32010

TMS32010 seperti keluarga TMS320 lainnya menggunakan arsitektur Harvard yang dimodifikasi (modified Harvard architecture). Pada arsitektur ini memori program dan data terletak pada ruang yang terpisah, sehingga memungkinkan overlap sepenuhnya antara fetch instruksi dan eksekusi. Dengan arsitektur ini memungkinkan adanya transfer antara ruang program dan data, sehingga koefisien-koefisien yang disimpan pada memori program dapat dipindahkan ke dalam RAM tanpa membutuhkan koefisien pada ROM secara terpisah. TMS32010 juga menyediakan instruksi-instruksi immediate dan subroutine-subroutine yang didasarkan pada harga-harga yang dihitung.

TMS32010 berisi ALU dan accumulator 32 bit untuk melakukan operasi aritmatik double-precision dan komplement kedua. ALU merupakan unit aritmatik untuk keperluan umum yang bekerja dengan word 16 bit yang diambil dari RAM data atau didapat dari instruksi immediate. ALU juga dapat melakukan operasi boolean, sehingga mampu melakukan manipulasi bit yang diperlukan pada kontroler kecepatan tinggi. Accumulator menyimpan

output dari ALU. Accumulator memiliki panjang 32 bit dan terbagi menjadi high-order word (bit 31-16) dan low-order word (bit 15-0).

Multiplier melakukan operasi perkalian 16x16 bit komplement kedua dengan hasil 32 bit dalam satu cycle instruksi. Multiplier terdiri dari tiga elemen yaitu T Register, P Register dan multiplier array. T Register 16 bit menyimpan bilangan yang akan dikalikan (multiplier) untuk sementara; P Register menyimpan hasil kali 32 bit. Harga-harga multiplier dapat diambil dari memori data atau diperoleh langsung dari instruksi MPYK (Multiply immediate). Hardware Multiplier on-chip yang cepat ini memungkinkan device melakukan operasi-operasi dasar dalam pengolahan sinyal digital seperti konvolusi, korelasi dan proses filter (filtering) dengan cepat dan efisien.

Dua buah shifter disediakan untuk manipulasi data. ALU Barrel shifter dapat melakukan geser kiri (left-shift) dari 0 hingga 16 pada word memori data yang diisikan ke ALU. Shifter ini memperluas high-order bit pada word data dan mengisi harga nol pada low-order bit untuk aritmatik komplement kedua. Accumulator parallel shifter melakukan geser kiri sebanyak 0, 1 atau 4 kali pada seluruh isi accumulator dan meletakkan hasil bit-bit high-order accumulator ke dalam RAM data. Kedua shifter tersebut berguna untuk operasi scaling dan ekstraksi bit.

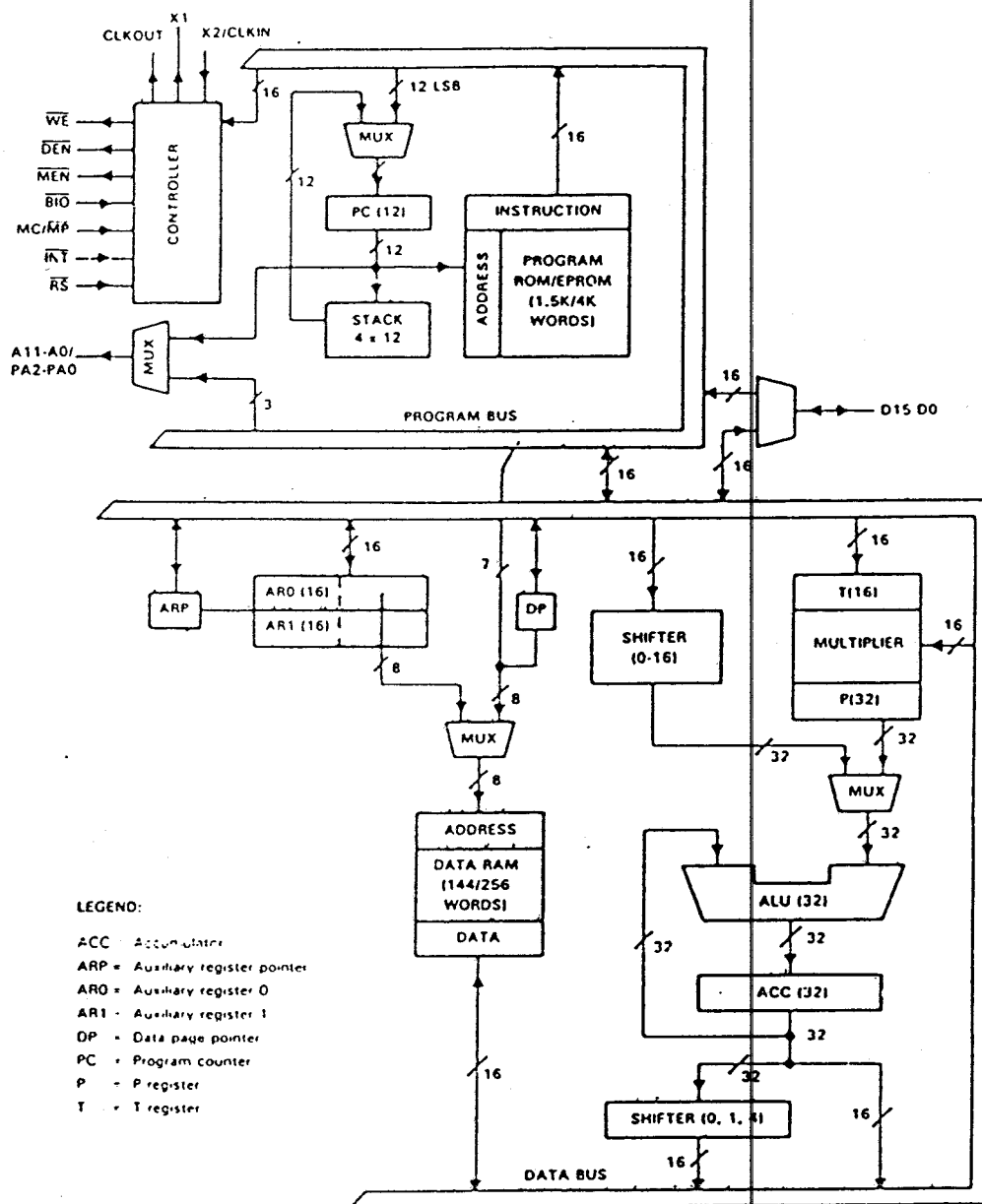
TMS32010 memiliki 144 word on-chip data RAM dan 1,5K word on-chip program ROM. TMS32010 ini dapat dioperasikan dalam dua mode operasi yang ditentukan oleh pin MC/ $\overline{\text{MP}}$ . Pada mode mikrokomputer 1,5K word on-chip program ROM dapat dipakai dan 2,5K word sisanya adalah memori eksternal. Sedangkan dalam mode mikro-prosesor, 4K word memori program adalah eksternal.

TMS32010 mempunyai hardware stack 4 level untuk menyimpan isi program counter selama terjadi interrupt dan subroutine call.

Diagram blok TMS32010 ditunjukkan pada gambar 3-2. Keterangan untuk tiap bagian pada gambar tersebut adalah sebagai berikut:

<u>UNIT</u>	<u>SIMBOL</u>	<u>FUNGSI</u>
Accumulator	ACC	Accumulator 32 bit terbagi menjadi high-order word (bit 31-16) dan low-order word (bit 15-0), digunakan untuk menyimpan output dari ALU.
Arithmetic Logic Unit	ALU	Adalah unit logika aritmatik 32 bit 2's komplement, mempunyai dua input 32 bit dan satu output 32 bit yang digunakan untuk mengisi accumulator.
Auxiliary Registers	ARO, AR1	Dua register 16 bit yang digunakan untuk data memory addressing dan loop count control. Kesembilan bit terbawah pada tiap register dapat digunakan untuk up/down counter.



Gambar 3-2<sup>19)</sup>

## DIAGRAM BLOK TMS32010

Auxiliary Register Pointer      ARP

Bit status yang menunjukkan auxiliary register mana yang sedang aktif.

19) Ibid., p. 3-5

Central Arithmetic Logic Unit	CALU	Gabungan dari ALU, multiplier, accumulator dan shifter.
Data Bus	D(15-0)	Bus 16 bit yang digunakan untuk jalur data dari RAM.
Data Memory Page Pointer	DP	Bit status yang menunjuk data page mana dari RAM yang sedang digunakan.
Data RAM		Terdiri dari 144 word untuk menyimpan data atau operand.
External Address Bus	A(11-0)/PA(2-0)	Bus 12 bit yang digunakan untuk meng-address program memori eksternal. Ketiga bit terendah adalah address port dalam mode I/O.
Interrupt Flag	INTF	Flag satu bit yang menunjukkan adanya interrupt request.
Interrupt Mode	INTM	Bit status untuk ne-mask interrupt flag.
Multiplier	MULT	Parallel hardware multiplier 16x16 bit.
Overflow Flag	OV	Bit status flag yang menunjukkan adanya overflow dalam operasi aritmatik.
Overflow Mode	OVN	Bit status yang menentukan mode saturated atau unsaturated dalam operasi aritmatik.
P Register	P	Register 32 bit yang berisi hasil kali dari operasi perkalian.
Program Bus	P(15-0)	Bus 16 bit yang digunakan untuk instruksi dari memori program.

Program Counter	PC(11-0)	Register 12 bit yang digunakan untuk meng-address memori program. PC selalu berisi address dari instruksi berikutnya yang akan dieksekusi. Isi PC di-update setelah operasi pen-decode-an instruksi selesai.
Program ROM		1,5K word on-chip ROM yang berisi kode program.
Shifter		Ada dua shifter. ALU barrel shifter melakukan geser kiri sebanyak 0 hingga 16 kali pada word memori data yang diisikan ke ALU. Accumulator parallel shifter melakukan geser kiri sebanyak 0, 1 atau 4 kali pada seluruh isi accumulator dan menyimpan hasil high-order word ke RAM.
Stack		Hardware stack 4x12 digunakan untuk menyimpan isi PC selama terjadi interrupt dan subroutine call.
Status	ST	Register status 16 bit yang berisi bit-bit status dan kontrol.
T Register	T	Register 16 bit yang berisi multi-plicand selama operasi perkalian.

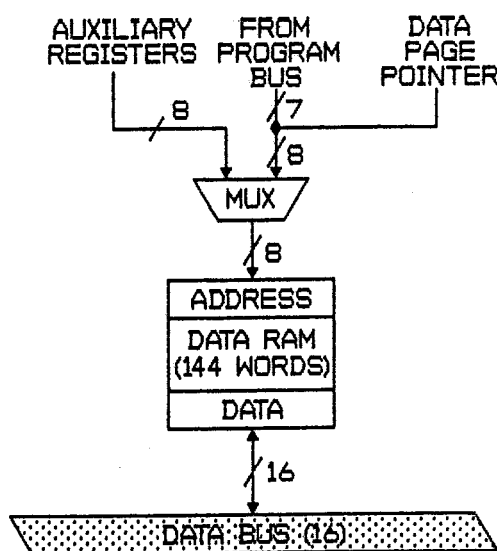
### III.3 ORGANISASI MEMORI

TMS32010 menggunakan arsitektur Harvard dimana memori data dan memori program terletak pada dua ruang

yang terpisah.

### III.3.1 Memori Data

Memori data terdiri dari 144 word 16 bit on-chip RAM seperti ditunjukkan pada gambar 3-3. Untuk ekspansi memori data, data bisa disimpan pada off-chip RAM kemudian dibaca ke on-chip RAM. Dua pasangan instruksi, TBLR/TBLW dan IN/OUT, disediakan untuk melakukan hal di atas. Instruksi TBLR (table read) dapat mentransfer isi memori program, baik on-chip maupun off-chip ROM/RAM, ke on-chip data RAM. Instruksi TBLW (table write) mentransfer isi RAM data ke off-chip program RAM.



Gambar 3-3<sup>20)</sup>

#### ON-CHIP DATA RAM

Kedua instruksi tersebut memerlukan waktu eksekusi tiga cycle. Apabila menggunakan instruksi IN/OUT, instruksi IN membaca data dari peripheral dan ditransfer ke RAM

<sup>20)</sup> Ibid., p. 3-11

data. Dengan menggunakan beberapa perangkat tambahan, instruksi IN dan OUT dapat digunakan untuk membaca dan menulis data dari RAM data ke peralatan penyimpanan eksternal yang sudah ditentukan alamatnya. Metoda ini lebih cepat karena instruksi IN dan OUT hanya memerlukan waktu eksekusi dua cycle.

### III.3.2 Memori Program

Program memori terdiri dari 1,5K word pada device. On-chip program ROM memungkinkan eksekusi program pada kecepatan penuh tanpa memerlukan memori program eksternal kecepatan tinggi.

Operasi memori program dapat dipilih dengan mengatur pin MC/ $\overline{\text{MP}}$  (microcomputer/mikroprosesor). Bila pin tersebut diset high berarti mode mikrokomputer. Sedangkan bila diset low berarti mode mikroprosesor.

Dalam mode mikrokomputer hanya disediakan lokasi 0 sampai 1523 dari ROM untuk pemakai. Sedangkan lokasi 1524-1535 digunakan untuk keperluan pengujian. Memori program sebanyak 2,5K dapat ditambahkan pada off-chip ROM. Sedangkan dalam mode mikroprosesor seluruh memori program 4Kx16 bit ditempatkan pada ROM eksternal.

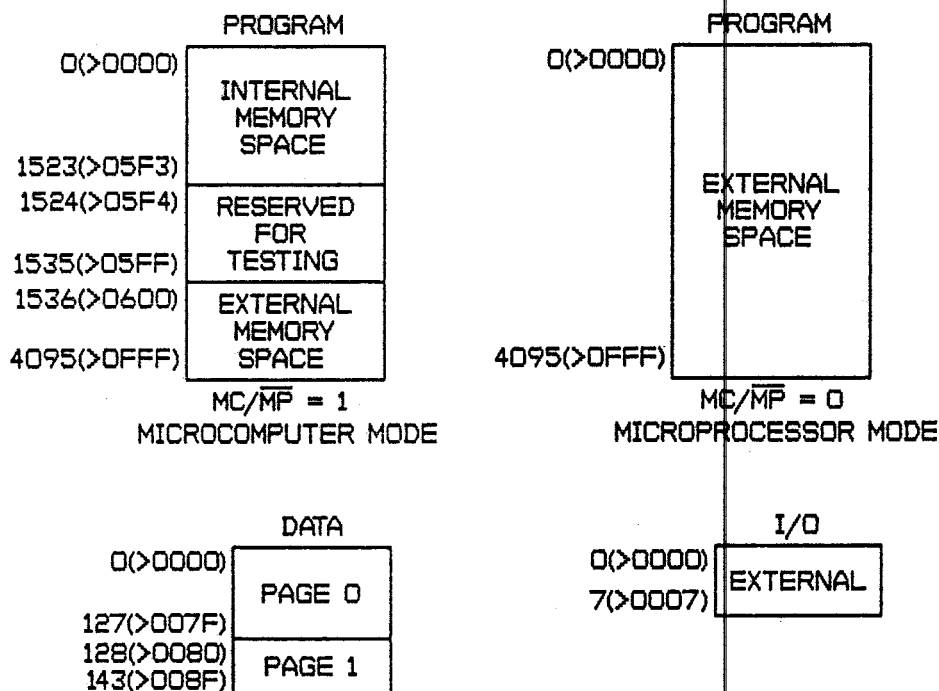
### III.3.3 Perpindahan Data

TMS32010 menyediakan instruksi untuk fungsi-fungsi perpindahan data pada on-chip RAM. Fungsi DMOV (data move) berguna dalam implementasi algoritma yang menggunakan operasi delay  $z^{-1}$ , seperti konvolusi dan filter digital.

Fungsi DMOV memungkinkan suatu word isi lokasi memori data yang sedang di-address dalam on-chip RAM disalin ke lokasi yang lebih tinggi, sementara data pada lokasi yang di-address dioperasikan pada cycle yang sama (misalnya pada instruksi CALU). Instruksi LTD (isi Register T, jumlahkan dengan hasil kali sebelumnya dan pindahkan data) juga menggunakan fungsi DMOV.

### III.3.4 Penetaan Memori

TMS 32010 memberikan tiga ruang alamat terpisah untuk memori program, memori data dan I/O seperti pada gambar 3-4. Memori program dikonfigurasi sesuai dengan kondisi pin  $\overline{MC}/\overline{MP}$ .



Gambar 3-4<sup>21)</sup>

### PEMETAAN MEMORI UNTUK TMS32010

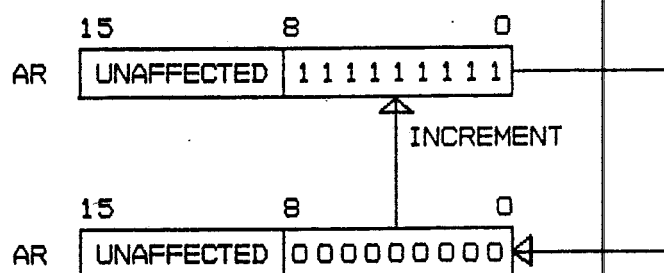
21) *Ibid.*, p. 3-13

### III.3.5 Auxiliary Register

TMS32010 mempunyai dua auxiliary register 16 bit (ARO dan AR1). Pada bagian ini akan dibahas masing-masing mengenai fungsi register dan bagaimana auxiliary register ini dipilih, diisi dan disimpan isinya.

Auxiliary register dapat digunakan untuk indirect addressing dari memori data, penyimpanan data sementara dan sebagai loop control. Indirect addressing dilakukan dengan meletakkan address memori data dari operand suatu instruksi pada kedelapan bit terendah auxiliary register. Register dipilih dengan single-bit Auxiliary Register Pointer (ARP) yang diisi harga 0 atau 1, yang masing-masing menunjukkan ARO atau AR1. ARP ini merupakan bagian dari register status dan dapat disimpan di memori.

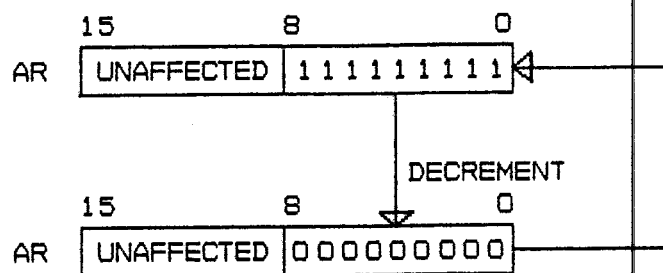
Apabila isi auxiliary register di-autoincrement/ autodecrement dengan instruksi indirect addressing atau dengan instruksi BAZ (branch on auxiliary register not zero), kesembilan bit terendah akan berubah (lihat gambar 3-5 dan gambar 3-6).



Gambar 3-5<sup>22)</sup>

INDIRECT ADDRESSING AUTOINCREMENT.

22) *Ibid.*, p. 3-15

Gambar 3-6<sup>23)</sup>

## INDIRECT ADDRESSING AUTODECREMENT.

Ketujuh bit teratas dari auxiliary register (bit 9 -15) tidak terpengaruh oleh operasi autoincrement/autodecrement termasuk autoincrement dari 11111111 (kesembilan bit terendah menuju 00000000) dan autodecrement dari 00000000 (kesembilan bit terendah menuju 11111111).

Auxiliary register dapat disimpan ke dan diisi dari memori data dengan instruksi SAR (store auxiliary register) dan LAR (load auxiliary register). SAR dan LAR mentransfer seluruh isi 16 bit ke dan dari auxiliary register meskipun indirect addressing dan loop counting hanya memakai sebagian dari auxiliary register.

Instruksi BANZ (branch on auxiliary not zero) memanfaatkan auxiliary register untuk dipakai sebagai loop counter. BANZ mengecek apakah isi register sama dengan nol. Jika tidak, maka isinya di-decrement dan melakukan branch.

23) Loc. cit.

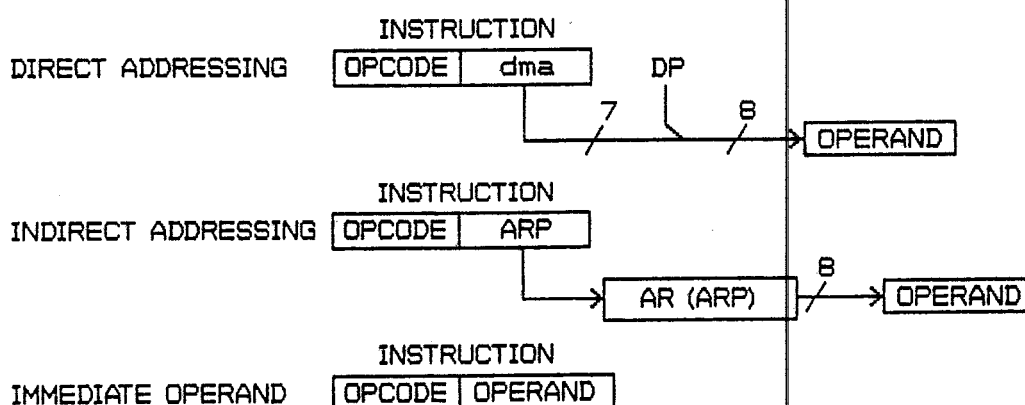


### III.3.6 Mode Memory Addressing

TMS32010 dapat meng-address sampai dengan 4K word pada memori program dan 144 word pada memori data. Ada tiga bentuk addressing yang dapat digunakan yaitu: direct, indirect dan immediate. Ketiga mode tersebut digambarkan seperti gambar 3-7.

Pada mode direct addressing, 1 bit data memory page pointer (DP) memilih page 0 yaitu lokasi memori 0-127 atau page 1 yaitu lokasi memori 128-143. Data memory address (dma) yang ditentukan oleh ketujuh LSB dari instruksi dan dirangkai dengan DP, meng-address word yang dikehendaki dalam page yang telah ditentukan. DP ini adalah bagian dari register status dan dapat disimpan di memori data.

Indirect addressing mempergunakan kedelapan bit terendah dari auxiliary register sebagai address memori



Gambar 3-7<sup>24)</sup>

METODA-METODA ADDRESSING PADA TMS32010

24) Ibid., p. 3-16

data yang cukup untuk meng-address 144 word data, tanpa memerlukan penentuan page. Auxiliary register dipilih dengan auxiliary register pointer (ARP). Auxiliary register dapat dibuat autoincrement/decrement selama suatu instruksi indirect addressing diberikan. Increment/decrement ini terjadi setelah instruksi yang sedang berlangsung selesai dieksekusi.

Bila digunakan mode immediate, operand dimasukkan ke dalam word instruksi itu sendiri.

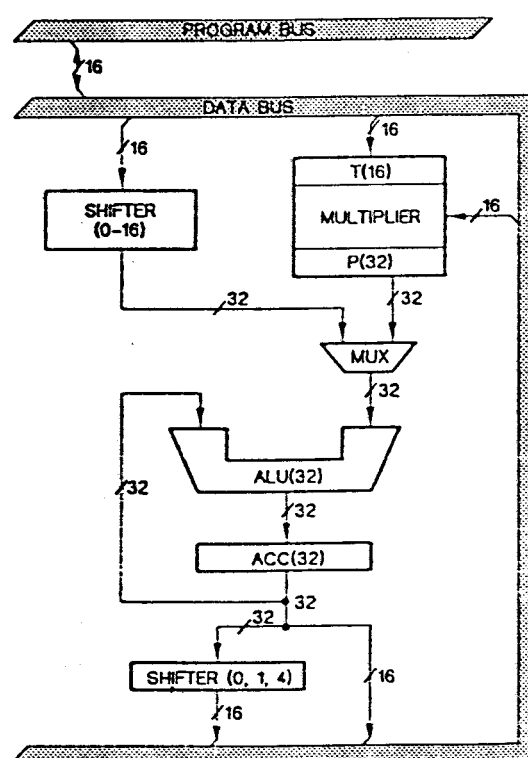
#### III.4 CENTRAL ARITHMETIC LOGIC UNIT (CALU)

CALU berisi parallel multiplier 16x16 bit, Arithmetic Logic Unit (ALU) 32 bit, accumulator (ACC) 32 bit dan dua shifter. Pada bagian ini akan dijelaskan mengenai komponen-komponen dan fungsi dari CALU. Diagram blok CALU ditunjukkan pada gambar 3-8.

Langkah-langkah berikut terjadi dalam implementasi dari suatu contoh operasi ALU :

1. Data diambil dari RAM melalui bus data.
2. Data dilewatkan melalui barrel shifter yang selanjutnya dapat digeser kiri sebanyak 0-16 bit, tergantung pada harga yang ditentukan oleh instruksi.
3. Data masuk ke ALU yang selanjutnya dioperasikan dan diisikan ke accumulator.
4. Hasil yang diperoleh di accumulator dilewatkan parallel left-shifter yang ada di output accumulator untuk membantu dalam scaling hasil.
5. Hasilnya disimpan di RAM data. Karena panjang

accumulator adalah 32 bit, maka isinya akan disimpan terpisah menjadi dua bagian yang sama. Input ALU selalu diberikan dari accumulator, dan input lainnya bisa berasal dari P Register dari multiplier atau dari barrel shifter yang diisi dari memori data.



Gambar 3-8<sup>25)</sup>

CENTRAL ARITHMETIC LOGIC UNIT (CALU).

### III.4.1 Shifter

Dua buah shifter disediakan untuk manipulasi data yaitu barrel shifter untuk menggeser data yang berasal

<sup>25)</sup> Ibid., p. 3-17

dari RAM data untuk diisikan ke ALU, dan parallel shifter untuk menggeser isi accumulator yang akan disimpan ke RAM data (lihat gambar 3-8).

Barrel shifter memiliki input 16 bit yang terhubung ke bus data dan output 32 bit yang terhubung ke ALU. Barrel shifter melakukan geser kiri sebanyak 0 hingga 16 bit pada seluruh word memori data yang dimasukkan ke dalamnya, dikurangkan dari, atau ditambahkan ke accumulator dengan instruksi LAC, SUB dan ADD. Dengan arithmetic left-shift, shifter mengisi harga nol pada LSB dan memberi tanda perluasan pada word memori data 16 bit menjadi 32 bit (misalnya bit di sebelah kiri MSB pada word data diisi 1 bila isi MSB sebelumnya berharga 1 atau diisi 0 bila MSB sebelumnya 0). Hal ini berbeda dengan logical left-shift yang selalu mengisi harga 0 di sebelah kiri MSB.

Parallel shifter hanya dapat diaktifkan dengan instruksi SACH (store high-order accumulator word). Instruksi ini mengakibatkan shifter diisi dengan 32 bit isi akumulator, kemudian data digeser ke kiri. Ke 16 bit MSB pada shifter disimpan di RAM, sehingga mengakibatkan hilangnya bit-bit high-order. Sedangkan Isi accumulator tetap tidak berubah. Parallel shifter hanya dapat melakukan pergeseran sebanyak 0, 1 atau 4. Pergeseran 1 dan 4 digunakan untuk operasi-operasi perkalian. Pergeseran ke kanan secara langsung tidak dapat dilakukan di sini.

### III.4.2 ALU dan Accumulator

ALU dan Akumulator 32 bit seperti gambar 3-8 melaksanakan fungsi-fungsi aritmatik dan logika dalam jangkauan yang luas, yang mayoritas dilakukan dalam satu clock cycle. Setiap kali suatu operasi dikerjakan di ALU, hasilnya ditransfer ke accumulator yang selanjutnya dapat dilakukan operasi lainnya seperti pergeseran. Data yang masuk ke ALU bisa diambil dari barrel shifter.

ALU adalah suatu unit logika aritmatik yang bekerja pada word data 16 bit, dan hasil operasinya 32 bit. ALU dapat menambah, mengurangi dan melakukan operasi-operasi logika. Accumulator selalu merupakan destinasi dan operand utama. Hasil operasi logika seperti pada tabel 3-1. Harga memori data (dma) adalah operand untuk setengah bagian bawah isi accumulator (bit 15-0). Nol adalah operand untuk setengah bagian atas isi accumulator.

Tabel 3-1. HASIL-HASIL OPERASI LOGIKA ACCUMULATOR<sup>26)</sup>

Fungsi	Acc bit 31-16	Acc bit 15-0
XOR	(0).XOR.(ACC (31-16))	(dma).XOR.(ACC (15-0))
AND	(0).AND.(ACC (31-16))	(dma).AND.(ACC (15-0))
OR	(0).OR.(ACC (31-16))	(dma).OR.(ACC (15-0))

Accumulator 32 bit menyimpan output dari ALU, yang juga sering merupakan input kembali untuk ALU. Accumulator dibagi dalam dua word 16 bit untuk penyim-

26) Ibid., p. 3-20

panan ke memori data yaitu high-order word (bit 31-16) dan low-order word (bit 15-0). Untuk melakukan penyimpanan high-order word dan low-order word isi accumulator ke memori data digunakan instruksi SACH dan SACL. Instruksi-instruksi ini dapat digunakan dalam implementasi double-precision arithmetic.

Accumulator juga mampu mensimulasikan efek saturasi dari sistem analog. Hal ini dapat dilakukan dengan menggunakan mode saturasi accumulator, yang dikontrol dengan bit register status OVM (overflow mode). Mode saturasi accumulator di-enable atau di-disable dengan menset atau mereset bit OVM melalui instruksi SOVM dan ROVM. Jika OVM diset dan operasi accumulator menghasilkan overflow, maka accumulator berisi bilangan paling positif atau paling negatif, tergantung pada tanda dari operand dan hasil yang didapat. Harga accumulator pada keadaan saturasi adalah 7FFFFFFF (positif) atau 80000000 (negatif). Jika OVM direset dan terjadi overflow, hasil overflow dimasukkan ke accumulator tanpa modifikasi. Sedangkan dalam operasi logika tidak dapat menghasilkan overflow.

Pengaktifan mode saturasi diperlukan terutama bila isi accumulator merupakan representasi harga suatu sinyal, karena bila mode saturasi tidak diaktifkan, overflow mengakibatkan diskontinuitas yang tidak diharapkan dalam representasi bentuk gelombang. Dalam keadaan mode saturasi, perilaku accumulator sangat mirip

dengan keadaan sistem analog yang mencapai batas maksimum atau saturasi bila diberi sinyal yang terlalu besar.

Apabila terjadi overflow, bit OV (overflow) pada register status diset, tidak peduli apakah bit OVM diset atau tidak. Instruksi BV (branch on overflow), yang melakukan branch hanya bila OV diset, memungkinkan program membuat keputusan berdasarkan kondisi apakah telah terjadi overflow atau tidak. Bila OV diset, maka OV hanya dapat direset dengan instruksi BV atau dengan mengisi langsung register status. Karena OV adalah bagian dari register status maka kondisinya dapat disimpan dalam memori data dengan instruksi SST (store status register) atau dapat diisi dengan instruksi LST (load status register).

TMS32010 ini juga dapat melakukan instruksi branch yang bergantung pada status ALU dan akumulator. Instruksi-instruksi seperti BLZ, BLEZ, BGEZ, BGZ, BNZ, BZ akan mengakibatkan terjadinya branch bila menemui kondisi tertentu.

#### III.4.3 Multiplier, T Register dan P Register

TMS32010 menyediakan hardware multiplier 16x16 bit yang mampu melakukan operasi perkalian dalam satu machine cycle dengan hasil kali 32 bit. Kedua register berikut mempunyai kaitan erat dengan multiplier, yaitu:

- Temporary Register (T) 16 bit yang menyimpan operand-operand untuk multiplier.

- Product Register (P) 16 bit yang menyimpan hasil kali.

Untuk menggunakan multiplier mula-mula operand harus diisikan ke T register melalui bus data dengan instruksi LT, LTA atau LTD. Kemudian instruksi MPY (multiply) atau MPYK (multiply immediate) memberikan operand kedua yang juga melalui bus data. Jika menggunakan instruksi MPY, maka harga pengali adalah bilangan 16 bit. Sedangkan jika menggunakan MPYK, harga pengali adalah konstanta immediate 13 bit yang berada dalam word instruksi MPYK. Setelah eksekusi selesai, hasil kali ditempatkan di P register. Hasil tersebut selanjutnya dapat ditambahkan, dikurangkan atau diisikan ke akumulator dengan instruksi PAC, APAC, SPAC, LTA atau LTD.

Isi P register tidak dapat diubah tanpa mengubah register-register yang lain. Terjadinya interrupt tidak dilayani hingga instruksi yang menyertai instruksi MPY/MPYK selesai dieksekusi. Oleh karena itu instruksi perkalian hendaknya selalu diikuti dengan instruksi yang menghubungkan P register dengan accumulator.

### III.5 SISTEM KONTROL

Sistem kontrol pada TMS32010 dilakukan oleh program counter dan stack, sinyal reset eksternal, interrupt dan register status.

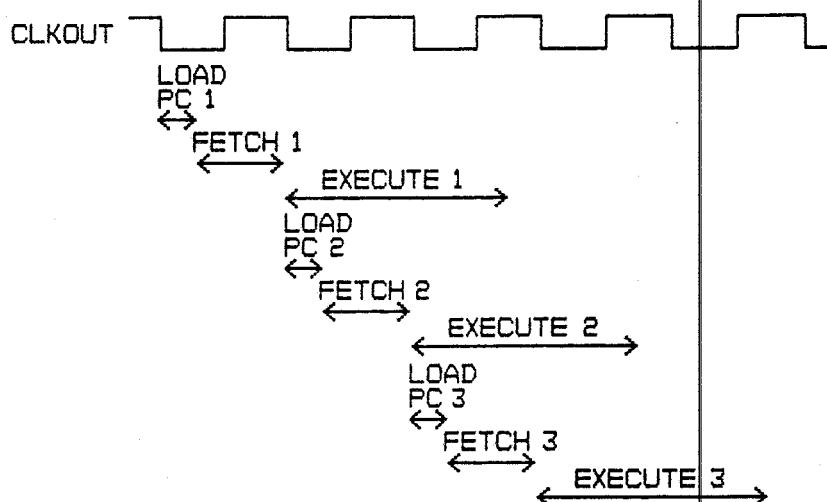


### III.5.1 Program Counter dan Stack

Program counter dan stack digunakan untuk eksekusi instruksi-instruksi branch, call subroutine, interrupt dan table read/table write. Program counter (PC) adalah register 12 bit yang berisi address memori program dari instruksi berikutnya yang akan dieksekusi. TMS32010 membaca instruksi dari lokasi memori program yang di-address oleh PC dan isi PC secara otomatis increment menyiapkan instruksi berikutnya yang akan diambil. PC diinisialisasi ke harga 0 dengan mengaktifkan pin reset (RS).

Seperti dikemukakan sebelumnya bahwa TMS32010 menggunakan arsitektur Harvard yang dimodifikasi. Pada arsitektur Harvard ruang memori program dan memori data terpisah, sehingga memungkinkan operasi pengambilan instruksi dan eksekusi dilakukan secara overlap. Gambar 3-9 memperlihatkan hal ini.

Memori program selalu di-address oleh isi PC. Isi PC dapat diubah oleh instruksi branch jika kondisi branch yang diuji adalah benar. Jika tidak, maka PC langsung increment ke address instruksi berikutnya. Semua branch di sini adalah absolut, tidak ada branch relative, jadi harga 12 bit yang didapat dari word instruksi branch diisi langsung ke PC agar melakukan branch. Bila terjadi interrupt atau instruksi subroutine call, isi PC di-push ke stack untuk menjaga kembalinya hubungan dengan konteks program sebelumnya.

Gambar 3-9<sup>27)</sup>

## ARSITEKTUR HARVARD

Stack mempunyai panjang 12 bit dengan kedalaman empat level. PC stack dapat digunakan pada instruksi PUSH dan POP. Instruksi PUSH memasukkan keduabelas LSB isi accumulator ke puncak stack (TOS). Setiap kali isi PC dimasukkan ke TOS, isi sebelumnya pada masing-masing level ditekan ke bawah sedangkan isi stack lokasi keempat (terbawah) sebelumnya akan hilang. Oleh karena itu data akan hilang apabila terjadi empat kali operasi PUSH berturut-turut tanpa diselingi oleh instruksi POP. POP merupakan hal yang sebaliknya. Instruksi POP mengeluarkan isi TOS dan masuk ke 12 bit LSB accumulator. Untuk menyimpan isi TOS ke memori data dapat digunakan instruksi SACL (store low-order accumulator) setelah instruksi POP. Dengan cara ini dapat dilakukan ekspansi stack ke RAM data. Dari RAM data, ia bisa di-copy ke RAM program eksternal dengan instruksi TBLW

<sup>27)</sup> *Ibid.*, p. 3-23

(table write). Dengan cara ini stack dapat diekspansi ke level yang sangat besar.

### III.5.2 Reset

Reset (RS) adalah non-maskable external interrupt yang dapat digunakan setiap saat. Reset secara normal diberikan setelah supply dinyalakan pada saat prosesor dalam keadaan tidak tentu (acak). Input reset low harus diberikan sedikitnya lima clock cycle.

Pada saat menerima sinyal RS, akan terjadi proses berikut:

1. DEN, WE, dan MEN menuju high.
2. Bus data D15-D0 berada pada keadaan high impedance.
3. PC berisi 0 dan A11-A0 juga berisi 0 pada clock cycle berikutnya setelah RS menuju low.
4. Interrupt di-disable dan interrupt flag register seluruhnya direset.

TMS32010 dapat direset selama waktu yang tak terbatas. Bit-bit status ARP, DP dan OVM tidak diinisialisasi oleh reset. Jadi ini bit-bit ini hendaknya bisa diinisialisasi dengan software setelah reset.

### III.5.3 Register Status

Register status terdiri dari lima bit status. Bit-bit status ini dapat diubah secara sendiri-sendiri dengan instruksi yang sesuai. Disamping itu instruksi SST melayani penyimpanan register status ke memori data. Instruksi LST mengisi register status dari memori

data kecuali bit INTM. Bit ini hanya bisa diubah dengan instruksi EINT/DINT (enable/ disable interrupt).

Tabel 3-2 memperlihatkan instruksi-instruksi yang mempengaruhi isi register status.

Isi register status bisa disimpan di memori data dengan instruksi SST. Jika instruksi SST dieksekusi dengan menggunakan mode direct addressing, device secara otomatis akan menyimpan informasi ini pada page 1 memori data pada lokasi yang telah ditentukan oleh instruksi. Apabila digunakan mode indirect addressing, isi register status bisa disimpan di sebarang lokasi RAM yang ditentukan oleh auxiliary register.

Instruksi SST tidak mengubah isi register status. Gambar 3-10 menunjukkan posisi bit-bit status yang akan tersimpan di lokasi RAM yang sesuai setelah eksekusi instruksi SST.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OV	OVM	INTM	1	1	1	1	ARP	1	1	1	1	1	1	0	DP

Gambar 3-10<sup>28)</sup>

#### ORGANISASI REGISTER STATUS

Instruksi LST bisa digunakan untuk mengisi register status. Instruksi LST tidak menganggap bit-bit status berada pada page 1, sehingga DP harus diset ke satu sebelum instruksi LST agar instruksi ini dapat mengakses bit-bit status yang disimpan pada page 1.

<sup>28)</sup> *Ibid.*, p. 3-26

Mode interrupt (INTM) tidak dapat diubah dengan instruksi LST, sedangkan bit-bit yang lain dapat diubah dengan instruksi ini.

Tabel 3-2. DEFINISI BIT-BIT REGISTER STATUS<sup>29)</sup>

Bit	Fungsi
ARP	Auxiliary Register Pointer. Merupakan bit tunggal untuk memilih AR yang akan digunakan pada indirect addressing. ARO = 0 memilih ARPO; ARP=1 memilih ARP1. ARP dapat diubah dengan menjalankan instruksi yang menggunakan indirect addressing, atau dengan instruksi LARP, MAR dan LST.
DP	Data Memory Page Pointer. Register DP satu bit yang dirangkai dengan tujuh LSB word instruksi untuk membentuk dma 8 bit. DP=0 memilih 128 word pertama memori data yaitu page 0. DP=1 memilih page 1, sisa word pada memori data. DP dapat diubah dengan instruksi LST, LDP dan LDPK.
INTM	Bit Interrupt Mode. Bila suatu interrupt sedang dilayani, bit INTM otomatis diset 1 sebelum memulai pelayanan routine interrupt. INTM=0 meng-enable semua maskable interrupt; INTM=1 mendisable semua maskable interrupt. INTM diset dan direset dengan instruksi DINT dan EINT. RS juga menset INTM. INTM tidak berpengaruh pada unmaskable RS interrupt. INTM tidak dipengaruhi oleh instruksi LST.
OV	Overflow flag. OV=0 menunjukkan bahwa accumulator tidak overflow. OV=1 menunjukkan bahwa terjadi overflow. OV tetap diset kecuali bila ada instruksi BV atau LST mereset OV.
OVM	Overflow Mode Bit. OVM=0 berarti mode overflow tidak aktif, menyebabkan hasil overflow tetap disimpan di accumulator tanpa mengalami perubahan. OVM=1 mengaktifkan mode overflow, mengakibatkan isi accumulator diset ke harga paling positif atau paling negatif tergantung nilai overflow yang didapat. Instruksi SOVM dan ROVM menset dan mereset bit ini. LST juga dapat digunakan untuk mengubah OVM.

29) Ibid., p. 3-25

### III.6 FUNGSI INPUT/OUTPUT

TMS 32010 melakukan bermacam-macam fungsi I/O untuk berhubungan dengan peralatan luar. Bus data paralel 16 bit dapat dimanfaatkan untuk melakukan fungsi-fungsi I/O dalam dua clock cycle dengan instruksi IN dan OUT. Port-port I/O diberi address oleh ketiga LSB bus address (PA2-PA0). Pengambilan input untuk operasi-operasi tes bit dan branch ( $\overline{\text{BIO}}$ ) dan input interrupt ( $\overline{\text{INT}}$ ) diberikan untuk memberikan fleksibilitas pada sistem.

Desain I/O disederhanakan dengan memperlakukan I/O sama seperti memori. Peralatan I/O dipetakan ke dalam ruang address I/O menggunakan address eksternal prosesor dan bus-bus data yang sama dengan pemetaan memori.

Input/output data dari dan ke peralatan luar dilakukan dengan instruksi IN dan OUT. Data ditransfer melalui bus data 16 bit ke dan dari memori data dengan mengaktifkan dua strobe yang independen yaitu data enable ( $\overline{\text{DEN}}$ ) dan write enable ( $\overline{\text{WE}}$ ).

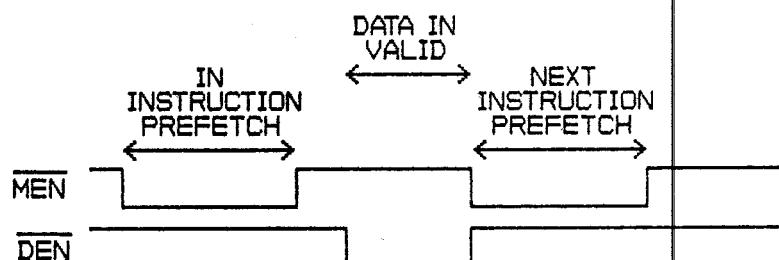
Bus data eksternal dua arah selalu dalam kondisi high impedance kecuali bila  $\overline{\text{WE}}$  aktif low.  $\overline{\text{WE}}$  menuju low selama cycle pertama instruksi OUT dan selama cycle kedua instruksi TBLW.

#### III.6.1 Operasi Input/Output

Ketiga pin port address (PA2-PA0) mengeluarkan address port selama instruksi OUT. Eksekusi instruksi IN membangkitkan strobe  $\overline{\text{DEN}}$  untuk mentransfer data dari

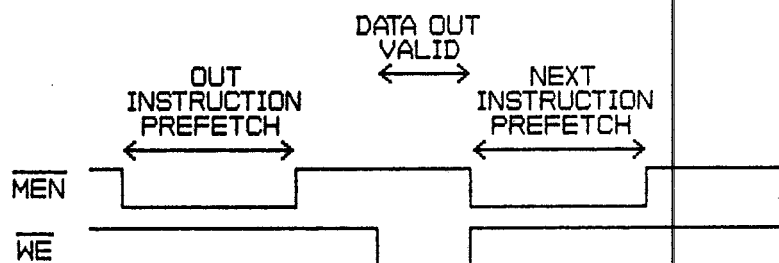
peripheral ke RAM data (lihat gambar 3-11). Instruksi IN merupakan satu-satunya instruksi yang menyebabkan  $\overline{\text{DEN}}$  aktif. Eksekusi instruksi OUT membangkitkan strobe  $\overline{\text{WE}}$  untuk mentransfer data dari RAM data ke peripheral (lihat gambar 3-12).  $\overline{\text{WE}}$  aktif bila ada instruksi OUT dan TBLW.

Sementara ketiga bus alamat LSB (PA2-PA0) digunakan sebagai alamat port oleh instruksi IN dan OUT, bit-bit yang lainnya (A13-A3) berada pada level "0" selama eksekusi dari instruksi-instruksi tersebut.



Gambar 3-11<sup>30)</sup>

#### DIAGRAM INSTRUKSI IN



Gambar 3-12<sup>31)</sup>

#### DIAGRAM INSTRUKSI OUT

30) *Ibid.*, p. 3-29

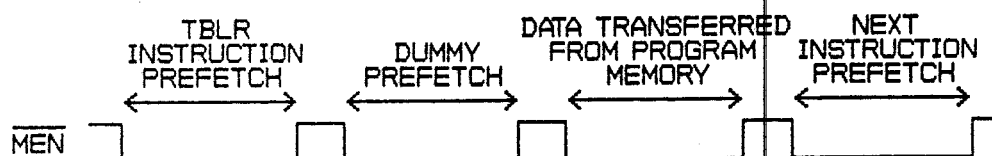
31) *Loc. cit.*

### III.6.2 Operasi Table Read/Table Write

Instruksi TBLR dan TBLW memungkinkan terjadinya transfer word data antara ruang program dan data. TBLR digunakan untuk membaca word data dari on-chip ROM atau off-chip program ROM/RAM ke RAM data. TBLW digunakan untuk menulis word data dari on-chip data RAM ke off-chip program RAM.

Eksekusi instruksi TBLR membangkitkan strobe  $\overline{MEN}$  untuk membaca word data dari memori program (lihat gambar 3-13). Eksekusi instruksi TBLW membangkitkan strobe  $\overline{WE}$  (lihat gambar 3-14). Bus data akan tetap diaktifkan dan sinyal  $\overline{WE}$  akan tetap dibangkitkan meskipun TMS32010 diset dengan mode microcomputer dan TBLW dilakukan pada lokasi program yang ada di on-chip ROM.

Dummy prefetch pada gambar 3-13 dan gambar 3-14 adalah pengambilan instruksi yang mengikuti instruksi TBLR atau TBLW dan tidak dipakai. Instruksi yang mengikuti TBLR atau TBLW diambil lagi pada akhir instruksi TBLR atau TBLW.

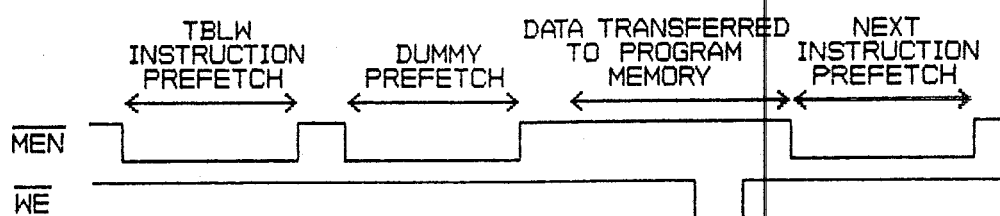


Gambar 3-13 32)

DIAGRAM INSTRUKSI TBLR

32) *Ibid.*, p. 3-30



Gambar 3-14<sup>33)</sup>

## DIAGRAM INSTRUKSI TBLW

Sinyal  $\overline{MEN}$ ,  $\overline{DEN}$  dan  $\overline{WE}$  masing-masing berdiri sendiri. Ada beberapa hal yang sangat penting untuk dipertimbangkan pada desain-desain yang menggunakan memori program. Karena instruksi OUT dan TBLW hanya menggunakan sinyal  $\overline{WE}$  untuk menunjukkan data yang berlaku, instruksi-instruksi tersebut tidak dapat dibedakan satu sama lain dalam basis sinyal interface. Bisa saja terjadi eksekusi instruksi TBLW akan menuliskan data ke peralatan luar, dan eksekusi instruksi OUT akan menindas memori program lokasi 0 hingga 7. Maka disarankan untuk menghindari pemetaan I/O dan RAM program eksternal keduanya pada lokasi 0 hingga 7.

## III.6.3 Pin I/O Keperluan Umum (BIO)

TMS32010 menyediakan pin untuk keperluan umum yang dikontrol dengan software. Pin BIO adalah input branch control untuk prosesor TMS32010.

Pin  $\overline{BIO}$  adalah pin eksternal yang menunjang operasi-operasi tes bit dan branch. Ketika input  $\overline{BIO}$  aktif low, eksekusi instruksi BIOZ mengakibatkan terjadinya

---

33) Loc. cit.

branch. Pin  $\overline{\text{BIO}}$  ini berguna untuk memantau status peralatan luar. Ini khususnya digunakan sebagai alternatif penggunaan interrupt bila loop yang memerlukan waktu kritis (time-critical loop) tidak dapat diganggu.

Untuk sistem yang menggunakan input tak sinkron dengan pin  $\overline{\text{BIO}}$  pada TMS32010 (NMOS), diperlukan perangkat eksternal untuk memastikan bahwa instruksi BIOZ dieksekusi dengan semestinya. Perangkat ini mensinkronkan sinyal input  $\overline{\text{BIO}}$  dengan rising edge dari CLKOUT TMS32010.

### III.7 INTERRUPT

TMS32010 menyediakan input interrupt eksternal untuk komunikasi dengan operasi-operasi eksternal yang kritis terhadap waktu. Interrupt dapat dibangkitkan dengan memberikan sinyal negative-going edge atau suatu sinyal level low pada pin input interrupt. Interrupt pada TMS32010 dapat di-mask (maskable) dengan menggunakan register status interrupt mode bit dan beberapa bit mask.

Untuk sistem yang menggunakan input tak sinkron pada pin interrupt ( $\overline{\text{INT}}$ ) TMS32010, diperlukan perangkat luar untuk memastikan proses interrupt bekerja dengan semestinya. Perangkat ini mensinkronkan sinyal input  $\overline{\text{INT}}$  dengan rising edge CLKOUT dari TMS32010.

Bila interrupt di-enable, interrupt menjadi aktif saat level low pada pin  $\overline{\text{INT}}$  atau bila negative-edge telah di-latch ke interrupt flag (INTF). Jika register

interrupt mode (INTM) diset ke nol, maka sinyal aktif interrupt pada interrupt internal prosesor akan diterima. Pelayanan interrupt dimulai dengan urutan-urutan berikut:

1. Interrupt diterima, bit INTF (interrupt flag) dire-set.
2. Bit INTM diset 1 untuk men-disable interrupt selanjutnya.
3. Isi PC yang ada dimasukkan ke TOS (top of stack).
4. PC yang baru diset pada 2.

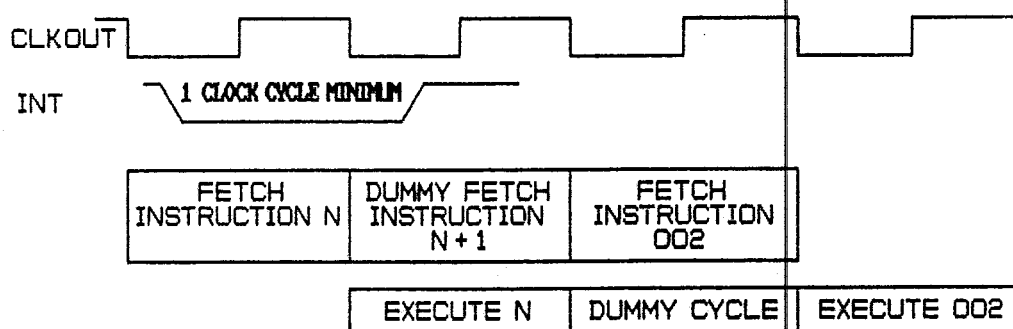
Pelayanan interrupt dimulai pada memori program address 2. Pada akhir pelayanan interrupt, instruksi EINT harus diberikan untuk mereset register INTM sehingga dapat menerima interrupt lagi. Instruksi DINT atau reset secara hardware juga akan menset register INTM pada 1 sehingga men-disable interrupt. Instruksi EINT harus diberikan untuk meng-enable kembali interrupt.

Pelayanan interrupt akan ditunda dalam hal-hal sebagai berikut:

1. Sampai berakhirnya seluruh cycle dari instruksi multicycle,
2. Sampai instruksi yang mengikuti instruksi MPY atau MPYK diselesaikan, atau
3. Sampai instruksi yang mengikuti instruksi EINT dieksekusi (bila interrupt sebelumnya sudah di-disable). Hal ini memungkinkan instruksi RET dapat

dieksekusi setelah interrupt di-enable pada akhir routine suatu interrupt.

Gambar 3-15 menunjukkan urutan instruksi yang terjadi saat interrupt aktif. Dummy fetch adalah instruksi yang diambil tetapi tidak dieksekusi. Instruksi ini akan diambil ulang dan dieksekusi setelah routine interrupt selesai dilaksanakan.



Gambar 3-15<sup>34)</sup>

#### DIAGRAM WAKTU INTERRUPT

### III.8 INSTRUKSI BAHASA ASSEMBLY TMS32010

TMS32010 memberikan set instruksi yang sangat menunjang dalam aplikasi pemrosesan sinyal dan aplikasi yang lain seperti kontrol dengan kecepatan tinggi. Set instruksi yang ditunjukkan pada tabel 3.3 umumnya terdiri dari satu cycle dan panjang instruksi satu word, sehingga memungkinkan mengeksekusi instruksi dengan kecepatan maksimum 6,25 MIPS. Pada tabel 3.4 tampak bahwa hanya instruksi-instruksi branch dan I/O yang memerlukan multicycle. Pada bagian ini akan di-

<sup>34)</sup> Ibid., p. 3-34

bahas mengenai instruksi bahasa assembly TMS32010, termasuk mode-mode addressing dan set instruksi.

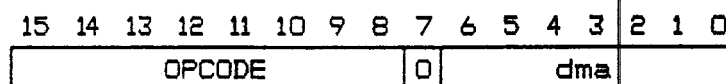
Seperti telah dijelaskan sebelumnya bahwa mode addressing yang dipakai oleh TMS32010 terbagi menjadi tiga, yaitu:

- Direct addressing,
- Indirect addressing, dan
- Immediate addressing.

### III.8.1 Mode Direct Addressing

Pada mode direct addressing, word instruksi pada ketujuh bit terendah berisi data memory address (dma). Field ini digabung dengan 1 bit register data memory page pointer (DP) membentuk data memory address 8 bit. Cara ini mengimplementasikan sistem paging dimana page pertama berisi 128 word dan page kedua berisi 16 word.

Direct addressing dapat digunakan oleh semua instruksi kecuali instruksi CALL, branch, immediate operand dan instruksi-instruksi tanpa operand. Format direct addressing ini ditunjukkan pada gambar 3-16 berikut:



Gambar 3-16<sup>35)</sup>

FORMAT INSTRUKSI MODE DIRECT ADDRESSING

<sup>35)</sup> Ibid., p. 4-3

Bit 15 hingga 8 berisi opcode. bit 7 = 0 yang menunjukkan mode addressing adalah direct. Bit 6 hingga 0 berisi data memory address (dma) yang dapat meng-address sampai dengan 128 word (1 page) memori data. Untuk meng-address semua ruang memori data diperlukan data memory page pointer.

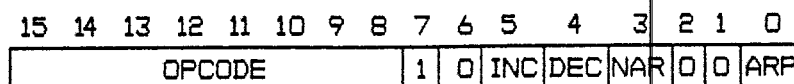
### III.8.2 Mode Indirect Addressing

Indirect addressing membentuk data memory address dari kedelapan bit terendah pada salah satu dari auxiliary register ARO dan AR1. Dengan indirect addressing ini cukup untuk mengaddress semua ruang memori data tanpa memerlukan penentuan page. Auxiliary Register Pointer (ARP) memilih auxiliary register yang akan dipakai. Auxiliary register dapat secara otomatis di-increment atau di-decrement secara paralel dengan eksekusi semua instruksi indirect, sehingga manipulasi tabel data dapat dilakukan dalam satu cycle. Increment/decrement terjadi setelah instruksi yang sedang berlangsung selesai dieksekusi.

Berikut ini adalah simbol-simbol yang digunakan dalam mode indirect addressing:

- \* Isi AR (ARP) digunakan sebagai data memory address.
- \*<sub>-</sub> Isi AR (ARP) digunakan sebagai address, kemudian decrement setelah mengakses memori data.
- \*<sub>+</sub> Isi AR (ARP) digunakan sebagai address, kemudian increment setelah mengakses memori data.

Format indirect addressing seperti pada gambar 3-17.



NOTE: NAR = new auxiliary register control bit

GAMBAR 3-17<sup>36)</sup>

### FORMAT INSTRUKSI MODE INDIRECT ADDRESSING

Bit 15 hingga 8 berisi opcode, bit 7 = 1 yang menunjukkan bahwa mode addressing adalah indirect. Bit 6 hingga 0 berisi bit-bit kontrol indirect addressing.

Bit 3 dan bit 0 mengontrol Auxiliary Register Pointer (ARP). Jika bit 3 = 0, isi bit 0 dimasukkan ke ARP setelah eksekusi instruksi yang bersangkutan. Jika bit 3 = 1, isi ARP tidak diubah. ARP = 0 menunjukkan bahwa isi ARO sebagai memory address. ARP = 1 menunjukkan bahwa isi AR1 sebagai memory address.

Bit 5 dan bit 4 mengontrol auxiliary. Jika bit 5 = 1, isi auxiliary register sedang digunakan di-increment dengan 1 setelah eksekusi. Jika bit 4 = 1 isi register yang sedang digunakan di-decrement dengan 1 setelah eksekusi. Jika bit 5 dan bit 4 keduanya 0, maka isi auxiliary register tidak di-increment ataupun di-decrement. Bit 6, 2 dan 1 harus selalu berharga 0.

### III.8.3 Mode Immediate Addressing

TMS32010 mempunyai lima instruksi immediate operand dimana immediate operand itu terdapat dalam word instruksi. Instruksi-instruksi itu dieksekusi dalam satu cycle. Panjang konstanta operand bergantung pada in-

<sup>36)</sup> Ibid., p. 4-4

struksinya. Instruksi immediate tersebut adalah:

LACK Load accumulator immediate (konstanta 8 bit)

LARK load auxiliary register immediate (konstanta 8 bit)

LARP Load auxiliary register pointer (konstanta 1 bit)

LDPK Load data memory page pointer immediate (konstanta 1 bit)

MPYK Multiply immediate (konstanta 13 bit).

#### III.8.4 Set Instruksi

Berikut ini akan diberikan daftar ringkasan set instruksi TMS32010. Instruksi-instruksi ini diklasifikasi menjadi 6 bagian yaitu:

- Instruksi untuk accumulator.
- Instruksi untuk auxiliary register dan data page pointer.
- Instruksi untuk perkalian, T register, dan P register.
- Instruksi branch dan call.
- Instruksi untuk kontrol.
- Instruksi untuk operasi I/O dan memori data.

Tabel 3.3 adalah daftar simbol dan singkatan yang digunakan dalam set instruksi TMS32010.

Tabel 3-3. SIMBOL-SIMBOL INSTRUKSI TMS32010<sup>37)</sup>

SIMBOL	ARTI	
A	Port address	
B	Branch address	

37) Ibid., p. 4-7



D	Field data memory address
I	Bit mode addressing
K	Field immediate operand
R	Field operand 1 bit untuk menentukan AR
S	Kode geser kiri 4 bit
X	Field left-shift accumulator 3 bit

Tabel 3.4. RINGKASAN SET INSTRUKSI TMS32010<sup>38)</sup>

ACCUMULATOR MEMORY REFERENCE INSTRUCTIONS						
Mnemonic and Description		Cycles	Words	MSB	16-Bit Opcode	
					MSB	LSB
ABS	Absolute value of accumulator	1	1	0111	1111	1000 1000
ADD	Add to accumulator with shift	1	1	0000	SSSS	1 DDD DDDD
ADDH	Add to high accumulator	1	1	0110	0000	1 DDD DDDD
ADDS	Add to low accumulator with sign-extension suppressed	1	1	0110	0001	1 DDD DDDD
AND	AND with accumulator	1	1	0111	1001	1 DDD DDDD
LAC	Load accumulator with shift	1	1	0010	SSSS	1 DDD DDDD
LACK	Load accumulator immediate short	1	1	0111	1110	KKKK KKKK
OR	OR with accumulator	1	1	0111	1010	1 DDD DDDD
SACH	Store high accumulator with shift	1	1	0101	1XXX	1 DDD DDDD
SACL	Store low accumulator	1	1	0101	0000	1 DDD DDDD
SUB	Subtract from accumulator with shift	1	1	0001	SSSS	1 DDD DDDD
SUBC	Conditional subtract	1	1	0110	0100	1 DDD DDDD
SUBH	Subtract from high accumulator	1	1	0110	0010	1 DDD DDDD
SUBS	Subtract from low accumulator with sign-extension suppressed	1	1	0110	0011	1 DDD DDDD
XOR	Exclusive-OR with low accumulator	1	1	0111	1000	1 DDD DDDD
ZAC	Zero accumulator	1	1	0111	1111	1000 1001
ZALH	Zero low accumulator and load high accumulator	1	1	0110	0101	1 DDD DDDD
ZALS	Zero accumulator and load low accumulator with sign-extension suppressed	1	1	0110	0110	1 DDD DDDD
AUXILIARY REGISTER AND DATA PAGE POINTER INSTRUCTIONS						
Mnemonic and Description		Cycles	Words	MSB	16-Bit Opcode	
					MSB	LSB
LAR	Load auxiliary register	1	1	0011	100R	1 DDD DDDD
LARK	Load auxiliary register immediate short	1	1	0111	000R	KKKK KKKK
LARP	Load auxiliary register pointer immediate	1	1	0110	1000	1000 000K
LDP	Load data memory page pointer	1	1	0110	1111	1 DDD DDDD
LDPK	Load data memory page pointer immediate	1	1	0110	1110	0000 000K
MAR	Modify auxiliary register	1	1	0110	1000	1 DDD DDDD
SAR	Store auxiliary register	1	1	0011	000R	1 DDD DDDD
T REGISTER, P REGISTER, AND MULTIPLY INSTRUCTIONS						
Mnemonic and Description		Cycles	Words	MSB	16-Bit Opcode	
					MSB	LSB
APAC	Add P register to accumulator	1	1	0111	1111	1000 1111
LT	Load T register	1	1	0110	1010	1 DDD DDDD
LTA	Load T register and accumulate previous product	1	1	0110	1100	1 DDD DDDD
LTD	Load T register, accumulate previous product, and move data	1	1	0110	1011	1 DDD DDDD
MPY	Multiply (with T register, store product in P register)	1	1	0110	1101	1 DDD DDDD
MPYK	Multiply immediate	1	1	100K	KKKK	KKKK KKKK
PAC	Load accumulator with P register	1	1	0111	1111	1000 1110
SPAC	Subtract P register from accumulator	1	1	0111	1111	1001 0000

38) Ibid., p. 4-9

Tabel 3.4. RINGKASAN SET INSTRUKSI TMS32010 (lanjutan)

BRANCH/CALL INSTRUCTIONS					
Mnemonic and Description		Cycles	Words	MSB	16-Bit Opcode LSB
B	Branch unconditionally	2	2	1111 0000	1001 0000 0000 0000
BANZ	Branch on auxiliary register not zero	2	2	1111 0000	0100 0000 0000 0000
BGEZ	Branch if accumulator $\geq 0$	2	2	1111 0000	1101 0000 0000 0000
BGZ	Branch if accumulator $> 0$	2	2	1111 0000	0000 0000 0000 0000
BIOZ	Branch on I/O status = 0	2	2	1111 0000	1100 0000 0000 0000
BLEZ	Branch if accumulator $\leq 0$	2	2	1111 0000	0110 0000 0000 0000
BLZ	Branch if accumulator $< 0$	2	2	1111 0000	0000 0000 0000 0000
BNZ	Branch if accumulator $\neq 0$	2	2	1111 0000	1010 0000 0000 0000
BV	Branch on overflow	2	2	1111 0000	1110 0000 0000 0000
BZ	Branch if accumulator = 0	2	2	1111 0000	0101 0000 0000 0000
CALA	Call subroutine indirect	2	1	0111	1111 1000 1100 0000
CALL	Call subroutine	2	2	1111 0000	1000 0000 0000 0000
RET	Return from subroutine	2	1	0111	0000 0000 0000 0000
CONTROL INSTRUCTIONS					
Mnemonic and Description		Cycles	Words	MSB	16-Bit Opcode LSB
DINT	Disable interrupt	1	1	0111	1111 1000 0001 0000
EINT	Enable interrupt	1	1	0111	1111 1000 0010 0000
LST	Load status register from data memory	1	1	0111	1011 1000 0000 0000
NOP	No operation	1	1	0111	1111 1000 0000 0000
POP	Pop top of stack to low accumulator	2	1	0111	1111 1001 1101 0000
PUSH	Push low accumulator onto stack	2	1	0111	1111 1001 1100 0000
ROVM	Reset overflow mode	1	1	0111	1111 1000 1010 0000
SOVM	Set overflow mode	1	1	0111	1111 1000 1011 0000
SST	Store status register	1	1	0111	1100 1000 1000 0000
I/O AND DATA MEMORY OPERATIONS					
Mnemonic and Description		Cycles	Words	MSB	16-Bit Opcode LSB
DMOV	Data move in data memory	1	1	0110	1001 1000 1000 1000
IN	Input data from port	2	1	0100	0AAA 1000 1000 1000
OUT	Output data to port	2	1	0100	1AAA 1000 1000 1000
TBLR	Table read	3	1	0110	0111 1000 1000 1000
TBLW	Table write	3	1	0111	1101 1000 1000 1000

## BAB IV

### PERENCANAAN

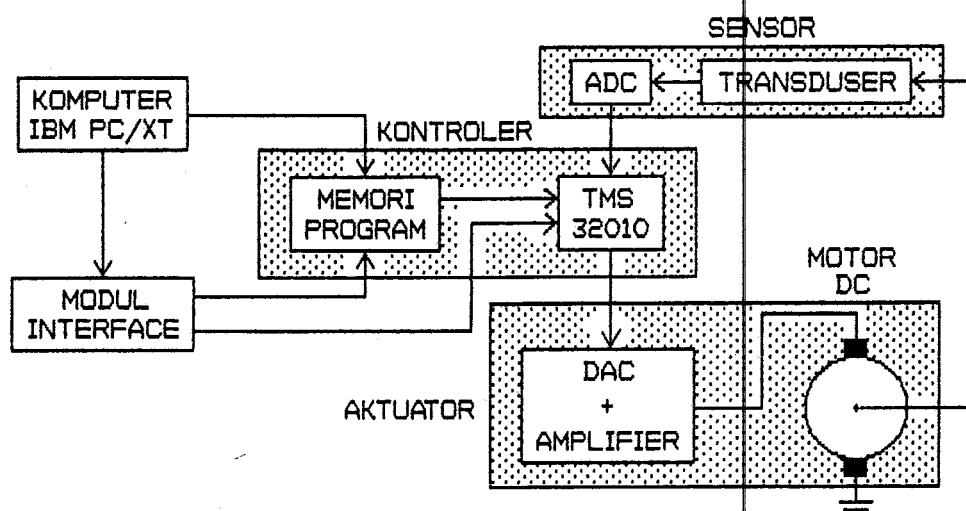
#### IV.1 PENDAHULUAN

Setelah mempelajari teori-teori tentang pengaturan suatu sistem serta komponen-komponennya maka gilirannya pada bab ini akan dibahas tentang perencanaan perangkat keras dan perangkat lunak dari sistem yang akan dibuat.

Seperti telah dijelaskan pada bab II bahwa dalam suatu sistem kontrol otomatis terdapat empat bagian utama yaitu *sensor*, *kontroler*, *aktuator* dan *plant* (peralatan yang dikontrol). Pada pengaturan posisi (disebut juga servomekanisme posisi) dengan motor DC ini digunakan potensiometer linier sebagai sensor posisi. Untuk aktuator yang berfungsi sebagai penggerak plant digunakan motor DC yang digerakkan oleh rangkaian penguat daya. Sedangkan kontroler yang berupa kontroler digital terdiri dari Prosesor TMS32010 sebagai unit pengolah data serta memori sebagai penyimpan program. Penghubung antara bagian analog dan digital digunakan konverter ADC dan DAC. ADC mengubah sinyal analog yang dihasilkan oleh sensor posisi menjadi data biner untuk selanjutnya diolah oleh prosesor. Sedangkan DAC mengubah data yang dihasilkan prosesor menjadi sinyal analog yang kemudian diperkuat oleh aktuator untuk menggerakkan motor sudut yang diinginkan.

Pada perencanaan sistem tersebut di atas ada sedikit persoalan dalam hal memasukkan instruksi ke

memori program yang nantinya akan dieksekusi oleh prosesor TMS32010. Hambatan ini karena tidak tersedianya fasilitas penunjang yang berupa program *assembler* untuk TMS32010 sehingga instruksi-instruksi tersebut harus diterjemahkan sendiri secara langsung ke kode heksadesimal. Akan tetapi kesulitan ini dapat diatasi dengan meng-interface-kan sistem kontroler dengan komputer IBM PC/XT. Fungsi utama komputer adalah sebagai pengirim "data", yang mana data tersebut merupakan kode-kode instruksi yang akan dieksekusi oleh prosesor TMS32010. Setelah tugas ini selesai, maka antara sistem kontroler dengan komputer hubungannya diputus secara elektronik. Keuntungan lain dengan cara interface ini adalah sumber tegangan yang diperlukan untuk mencatu sistem kontrol dapat diambil langsung dari IBM PC/XT. Diagram blok kontroler ditunjukkan pada gambar 4-1.



Gambar 4-1

## DIAGRAM BLOK SISTEM KONTROLER

## IV.2 SLOT PERIPHERAL IBM PC/XT

Sebelum menginjak lebih lanjut pada perencanaan sistem kontroler, baiklah lebih dahulu akan dibahas tentang slot peripheral pada IBM PC/XT, karena untuk dapat meng-interface-kan suatu sistem dengan komputer harus diketahui dengan pasti letak pin-pin dan fungsi masing-masing dari slot peripheral IBM PC/XT.

Seperti ditunjukkan pada gambar 4-2, slot peripheral IBM PC/XT terdiri dari 62 pin. Sinyal-sinyal tersebut dapat dijelaskan sebagai berikut:

### OSC (Oscillator)

Sinyal ini merupakan sinyal output dengan frekuensi 14,31818 MHz dan perioda sekitar 70 ns.

Duty cycle kira-kira 50%. Sinyal ini merupakan sinyal frekuensi tertinggi pada bus dan seluruh sinyal pada sistem dibangkitkan oleh sinyal ini.

### CLK (Clock)

Sinyal ini diambil dari sinyal OSC yang dibagi tiga sehingga mengeluarkan frekuensi 4,77 MHz. Sinyal CLK tidak simetris, melainkan mempunyai duty cycle sepertiga/duapertiga. Perioda sinyal ini 210 ns dengan waktu level high 70 ns dan waktu level low 140 ns.

### RESET DRV (Reset Driver)

Sinyal ini mengeluarkan sinyal aktif high selama proses system power-on dan tetap aktif sampai seluruh sistem mencapai level operasi yang ditentukan, kemudian sinyal ini tidak aktif (low).

A0-A19

Bit address A0-A19 digunakan untuk address memori dan I/O. Ke-20 sinyal ini diaktifkan oleh mikroprosesor 8088 untuk menunjuk pada address yang dituju selama read cycle atau write cycle memori dan I/O.

D0-D7

Kedelapan jalur data dua arah (bidirectional) ini digunakan untuk mentransfer data antara mikroprosesor 8088, memori dan I/O.

ALE (Address Latch Enable)

Sinyal output ini diaktifkan dari bus controller 8288. Ini digunakan untuk menunjukkan bahwa bus address berlaku untuk permulaan suatu bus cycle. Sinyal ini menjadi aktif high sesaat sebelum bus address stabil. Sinyal ini digunakan untuk melatch informasi alamat dari bus address/data lokal dari mikroprosesor 8088.

I/O CH CK (I/O Channel Check)

Sinyal input level low ini digunakan untuk memberitahukan kondisi error pada bus yang berasal dari interface card. Sinyal ini bila diset low akan membangkitkan nonmaskable interrupt (NMI) pada mikroprosesor 8088.

I/O CH RDY (I/O Channel Ready)

Sinyal input ini digunakan untuk memperpanjang waktu bus cycle, sehingga memori atau port I/O

yang memiliki respon lebih lambat daripada bus cycle normal empat clock (840 ns) masih dapat dihubungkan dengan bus sistem. Jika suatu memori atau port I/O perlu memperpanjang bus cycle, ia harus memberikan sinyal low pada I/O CH RDY ketika ia men-decode address tersebut dan menerima suatu instruksi MEMR, MEMW, IOR atau IOW. Sinyal ini harus dikontrol dengan hati-hati sehingga hanya diperlukan saat kondisi menunggu (wait state).

#### IRQ2 - IRQ7 (Interrupt Requests 2 through 7)

Keenam sinyal input ini digunakan untuk membangkitkan interrupt request ke mikroprosesor 8088 dari bus sistem. Sinyal-sinyal tersebut langsung menuju interrupt controller 8259A. Program BIOS menginisialisasi 8259A sehingga IRQ2 merupakan prioritas tertinggi dan IRQ7 terendah.

#### IOR (I/O Read)

Sinyal ini merupakan sinyal output aktif low, yang berasal dari bus controller 8288. Ini digunakan untuk memberitahu pada port I/O bahwa bus cycle yang dilakukan 8088 adalah read cycle dari port I/O dan address pada bus address merupakan address port I/O.

#### IOW (I/O Write)

Sinyal ini merupakan sinyal output aktif low. Ia juga diaktifkan oleh 8288 dan menunjukkan bahwa bus address merupakan address port I/O dan bus data berisi data yang akan ditulis ke port I/O.

MEMW

Sinyal ini aktif low digunakan untuk menulis data dari bus sistem ke memori. Sinyal ini digerakkan oleh bus controller 8288 yang menunjukkan bahwa bus address berisi address lokasi memori tempat data akan ditulis.

MEMR

Sinyal ini aktif low digunakan untuk membaca data dari memori. Ia juga digerakkan oleh 8288 yang menunjukkan bahwa bus alamat berisi address lokasi memori yang ditunjuk untuk dibaca.

DRQ1 - DRQ3 (Direct-memory Access Request 1 through 3)

Ketiga jalur ini adalah input aktif high digunakan oleh interface untuk DMA request. Jika suatu peralatan atau interface ingin mentransfer data antara dirinya dengan memori secara langsung tanpa melibatkan mikroprosesor 8088, maka DMA request dimulai dengan memberikan sinyal level high ke jalur DRQ. Jalur ini berhubungan langsung ke DMA controller 8237-5. Program pada ROM BIOS menginisialisasi sehingga DRQ1 merupakan prioritas tertinggi dan DRQ3 terendah. Sebenarnya yang memiliki prioritas tertinggi adalah DRQ0, akan tetapi ini tidak disediakan pada bus sistem. DRQ0 digunakan sistem komputer untuk me-refresh memori dinamik.



DACK0 - DACK3 (Direct-memory Access Acknowledge 0 through 3)

Keempat sinyal ini mempunyai output aktif low yang digunakan oleh 8237-5 untuk menunjukkan bahwa DRQ telah diterima dan DMA controller akan menggunakan bus serta meneruskan cycle DMA yang diminta.

AEN ( Address Enable)

Sinyal ini merupakan output aktif high yang digunakan oleh DMA controller logic. Ini menunjukkan bahwa DMA bus cycle sedang berlangsung. Sinyal ini juga digunakan untuk melepas hubungan bus-bus address, data dan kontrol mikroprosesor 8088 dari bus sistem dan mengaktifkan address dan bus kontrol dari DMA controller.

TC (Terminal Count)

Sinyal output ini aktif high digunakan oleh DMA controller 8237-5. Sinyal ini menunjukkan bahwa salah satu dari kanal DMA telah mencapai jumlah cycle transfer yang direncanakan. Sinyal ini khususnya digunakan untuk menghentikan transfer data blok DMA.

+5 V DC

Level tegangan +5 volt tersedia pada dua pin konektor slot dan mempunyai toleransi  $\pm 5\%$  (+4,75 hingga +5,25 volt dc).

+12 V DC

Level tegangan +12 volt tersedia pada satu pin dengan toleransi  $\pm 5\%$  (+11,4 hingga +12,6 volt dc).

SIGNAL	PIN	PIN	SIGNAL
GND	81	A1	IO CH CK
RESET DRV	82	A2	D7
+5 V DC	83	A3	D6
IRQ2	84	A4	D5
-5 V DC	85	A5	D4
DRQ2	86	A6	D3
-12 V DC	87	A7	D2
NOT USED <sup>39)</sup>	88	A8	D1
+12 V DC	89	A9	D0
GND	810	A10	IO CH RDY
MEMW	811	A11	AEN
MEMR	812	A12	A19
IOW	813	A13	A18
IOR	814	A14	A17
DACK 3	815	A15	A16
DRQ3	816	A16	A15
DACK 1	817	A17	A14
DRQ1	818	A18	A13
DACK 0	819	A19	A12
CLR	820	A20	A11
IRQ7	821	A21	A10
IRQ6	822	A22	A9
IRQ5	823	A23	A8
IRQ4	824	A24	A7
IRQ3	825	A25	A6
DACK 2	826	A26	A5
T/C	827	A27	A4
ALE	828	A28	A3
+5 V DC	829	A29	A2
OSC	830	A30	A1
GND	831	A31	A0

Gambar 4-2<sup>39)</sup>

## SLOT PERIPHERAL IBM PC/XT

39) Lewis C. Eggebrecht, 1987, Interfacing to the IBM Personal Computer, p.77

#### -5 V DC

Level tegangan -5 volt tersedia pada satu pin dengan toleransi  $\pm 10\%$  (-4,5 hingga -5,5 volt dc).

#### -12 DC

Level tegangan -12 volt tersedia pada satu pin dengan toleransi  $\pm 10\%$  (-10,8 hingga -13,2 volt dc).

#### GND (Ground)

Pin ground disediakan tiga pin pada konektor slot.

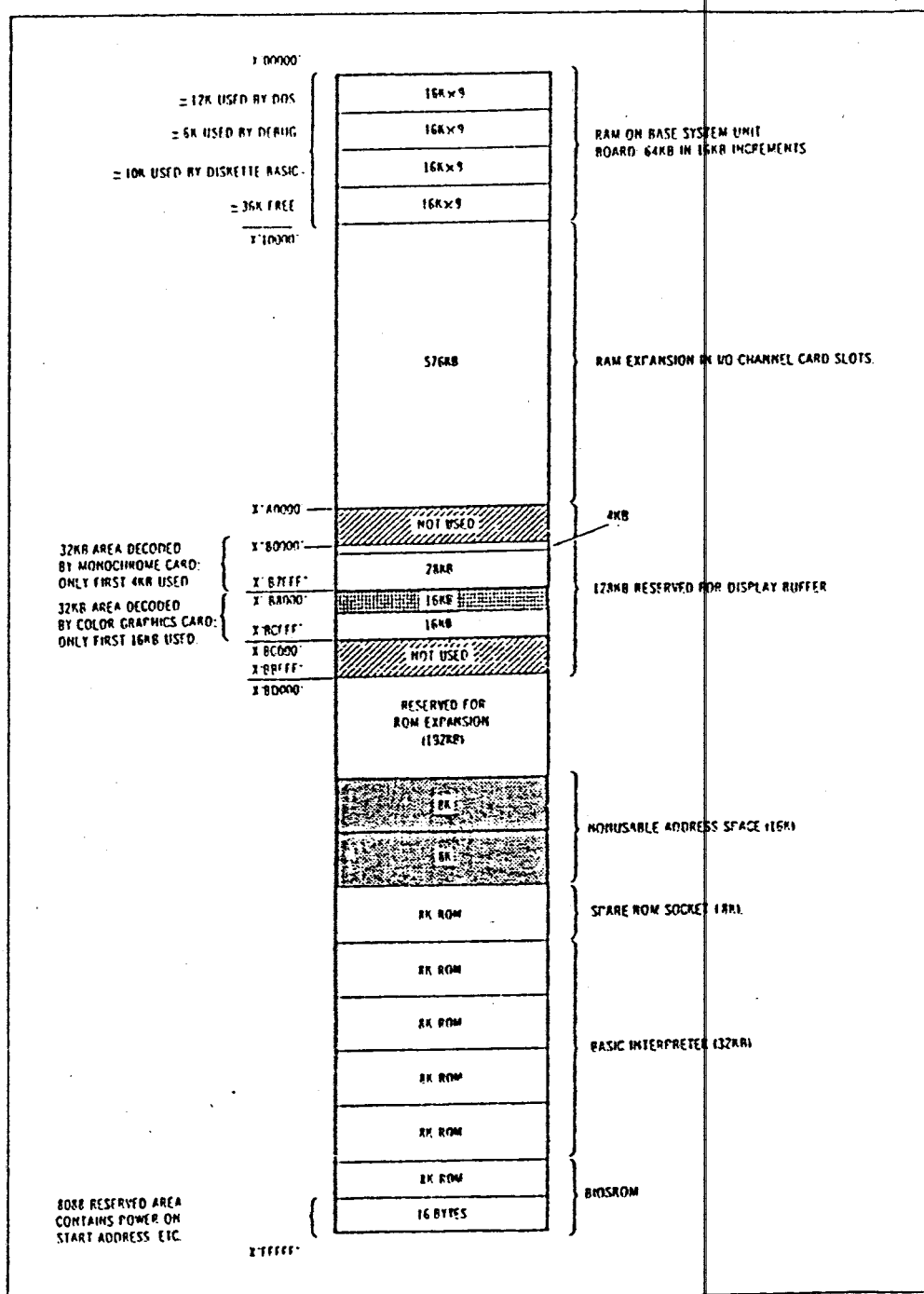
### IV.3 MODUL INTERFACE

Untuk menghubungkan antara komputer IBM PC/XT dengan kontroler diperlukan suatu rangkaian interface. Fungsi rangkaian ini adalah sebagai penghubung dan pemutus antara komputer dengan memori program seperti telah dijelaskan pada bagian pendahuluan di atas. Selain itu rangkaian ini juga digunakan untuk mereset prosesor TMS32010 selama berlangsungnya proses transfer data dari komputer memori program.

Alamat I/O pada perencanaan ini menggunakan pemetaan memori, maka harus ditentukan letak segmen memori komputer yang akan digunakan. Segmen ini harus merupakan segmen kosong, artinya tidak dipakai oleh sistem komputer. Pemetaan memori komputer IBM PC ditunjukkan pada gambar 4-3.

Penentuan segmen memori yang akan dipakai ini dilakukan oleh A16-A19 seperti tampak pada gambar 4-4.

Dengan menggunakan comparator 4-bit 74LS85 dan DIP-switch, alamat segmen dapat diubah-ubah.

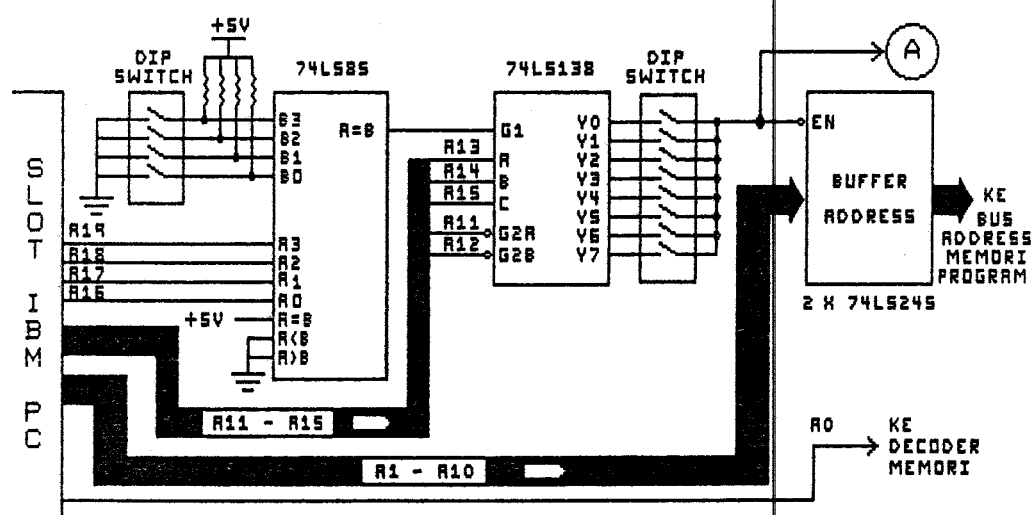


Gambar 4-3<sup>40)</sup>

### PEMETAAN MEMORI KOMPUTER IBM PC/XT

40) *Ibid.*, p.139

Bus address A1-A10 berhubungan dengan bus address memori program kontroler (A0-A9) melalui buffer address 74LS245. A0 digunakan untuk memilih salah satu memori program yang akan diisi data (akan dijelaskan pada bagian berikutnya). Bit-bit sisanya (A11-A15) digunakan untuk meng-enable decoder 74LS138. Output decoder yang aktif low digunakan untuk meng-enable buffer address.

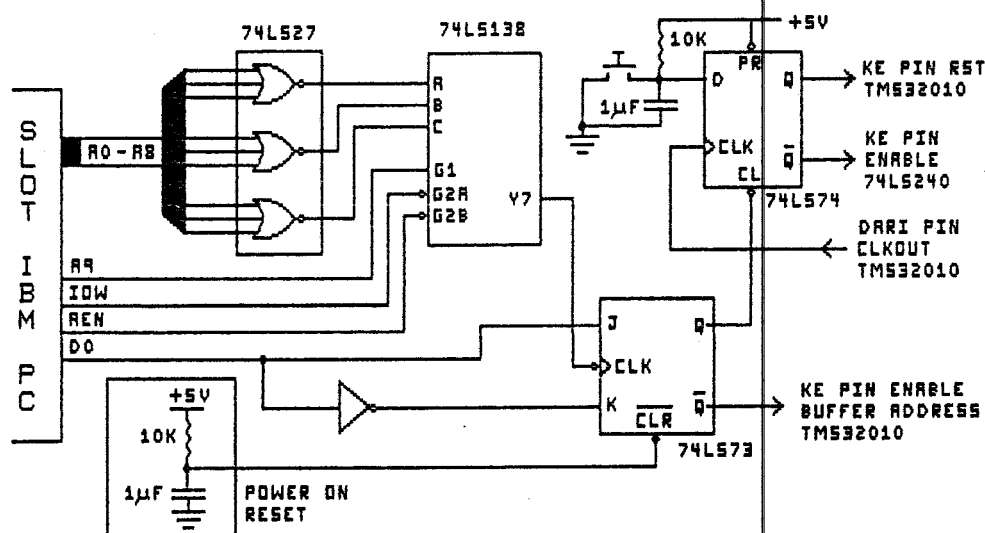


Gambar 4-4

## RANGKAIAN DECODER ADDRESS MEMORI PROGRAM

Selama proses transfer data berlangsung, hubungan memori program kontroler harus terlepas dari prosesor TMS32010. Ini dilakukan dengan memberikan data 0 pada port I/O address 200H. Data ini dipertahankan oleh JK flip-flop 74LS73 selama proses transfer data. Output Q dari JK flip-flop mereset D flip-flop. Dengan demikian TMS32010 berada dalam kondisi reset. Sedangkan output  $\bar{Q}$  JK flip-flop men-disable buffer address prosesor

TMS32010. Setelah transfer data selesai dilakukan maka port I/O address 200H diberi data 1 sehingga output JK flip-flop menjadi  $Q="1"$  dan  $\bar{Q}="0"$ . Kondisi ini membuat prosesor TMS32010 mulai bekerja dan bus address-nya terhubung dengan bus address memori program. Rangkaian ini ditunjukkan pada gambar 4-5.



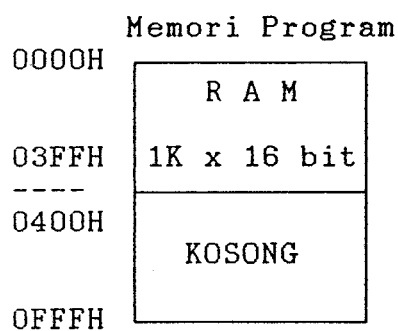
Gambar 4-5

#### RANGKAIAN ISOLASI TMS32010 DENGAN MEMORI PROGRAM

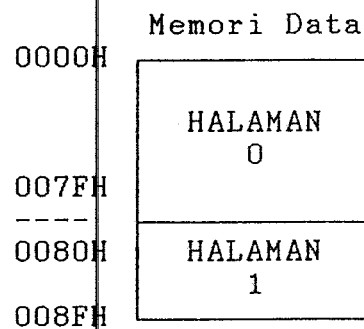
Bus data komputer IBM PC/XT ada 8-bit sedangkan prosesor TMS32010 memiliki 16-bit bus data. Untuk mengubah bus data dari 8-bit ke 16-bit dapat dilakukan dengan menggunakan dua buah buffer 74LS245 yang pada input-inputnya dihubungkan paralel, sedangkan outputnya dihubungkan dengan masing-masing bus data dari kedua memori program (lihat gambar 4-6). Buffer-buffer ini



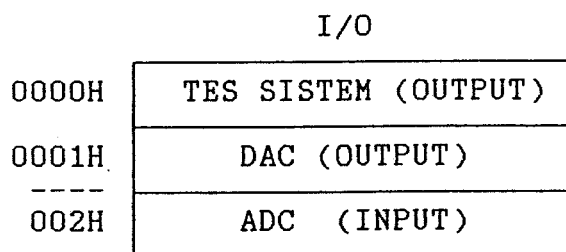
terpisah. Pada perencanaan sistem ini digunakan mode mikroprosesor yaitu dengan memberikan input '0' pada pin  $\overline{MC/MP}$ . Dengan demikian untuk memori program seluruhnya menggunakan off-chip program memory. Address yang dipakai sebesar 1K word (Gambar 4-7a).



(a)



(b)



(c)

Gambar 4-7  
PEMETAAN MEMORI dan I/O

Pemetaan untuk memori data menggunakan seluruh alamat memori data (144 word) on-chip RAM (Gambar 4-7b).

Sedangkan pada ruang I/O digunakan pemetaan seperti ditunjukkan pada gambar 4-7c sebagai berikut:

1. Address 0 : digunakan untuk mengaktifkan rangkaian tes sistem kontroler. Rangkaian ini berfungsi untuk



menguji apakah prosesor dan rangkaian output sudah bekerja dengan baik. Rangkaian tes ini terdiri dari sebuah data latch 8 bit 74LS373 dan 8 buah LED. Dengan mengirimkan data 8 bit pada address ini akan diketahui bahwa bila LED menyala sesuai dengan bit-bit data yang dikirimkan berarti prosesor dan rangkaian output bekerja dengan baik.

2. Address 1 : digunakan untuk mengirimkan data dari prosesor TMS32010 ke DAC melalui data latch 74LS374.
3. Address 2 : digunakan untuk mengambil data dari ADC yang merupakan hasil konversi tegangan analog yang didapat dari transduser potensiometer.

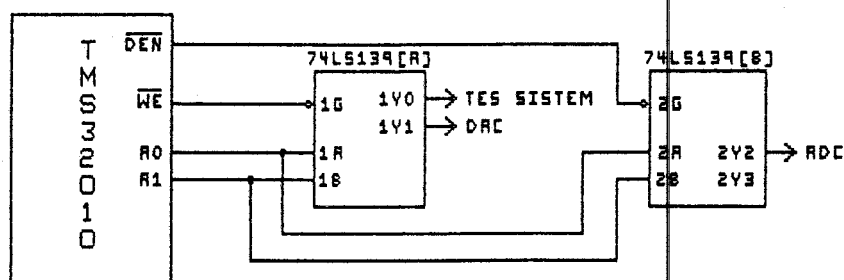
#### IV.4.2 Decoding

Decoding berfungsi untuk men-decode jalur address prosesor untuk mengaktifkan memori atau piranti I/O agar data yang dikirim ke atau diterima dari piranti tersebut tidak kacau.

Dari pemetaan memori di atas tampak bahwa untuk memori tidak perlu dilakukan decoding karena seperti telah dijelaskan bahwa ruang untuk memori program dan memori data memang terpisah. Dekoding diperlukan hanya pada address I/O yang nanti akan menentukan piranti I/O mana yang akan diaktifkan.

Pada gambar 4-8 ditunjukkan cara menghubungkan prosesor dengan memori program eksternal. Prosesor TMS32010 memiliki bus data 16 bit, sedangkan di pasaran tidak ada/sulit didapatkan memori yang memiliki bus





Gambar 4-9

## SISTEM PENDEKODEAN I/O

Output Y0-Y1 74LS139(a) meng-enable rangkaian output (dalam hal ini rangkaian tes sistem dan DAC), sedangkan output Y2 74LS139(b) meng-enable rangkaian input ADC.

## IV.4.3 Sistem Clock

Prosesor Sinyal Digital TMS32010 mempunyai dua pilihan sistem clock yaitu sistem clock internal dan sistem clock eksternal.

## - Sistem clock internal :

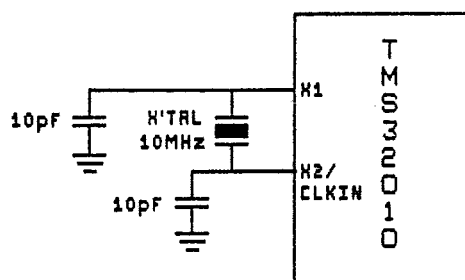
Clock internal diaktifkan dengan memasang sebuah osilator kristal pada pin X1 dan X2/CLKIN seperti ditunjukkan pada gambar 4-10. Frekuensi yang dihasilkan pada pin CLKOUT adalah 1/4 frekuensi kristal. Frekuensi kristal yang diperbolehkan berkisar antara 6,7 MHz - 20,5 MHz. Nilai Kapasitor C1 dan C2 masing-masing 10 pF.

## - Sistem clock eksternal

Sumber clock eksternal juga dapat digunakan dengan menghubungkan langsung sumber clock tersebut dengan

pin X2/CLKIN sedangkan pin X1 dibiarkan terbuka.

Pada perencanaan ini digunakan sistem clock internal dengan pertimbangan bahwa pemakaian sistem clock internal lebih mudah dilakukan dan menghemat pemakaian komponen daripada memakai clock eksternal. Frekuensi kristal yang dipakai adalah 10 MHz sehingga menghasilkan frekuensi CLKOUT = 2,5 MHz.

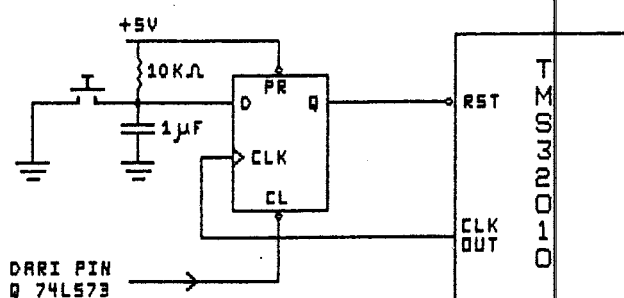


Gambar 4-10

#### SISTEM CLOCK INTERNAL

#### IV.4.4 Sistem Reset

Untuk mereset prosesor TMS32010 dapat diambil dari dua jalur, yaitu dari output Q JK flip-flop seperti dijelaskan pada bagian sebelumnya dan didapat dari penekanan tombol RESET seperti tampak pada gambar 4-11. Jalur reset yang diambil dari output JK flip-flop merupakan reset yang dilakukan dengan software pada setiap kali komputer akan mentransfer data ke memori program. Tombol RESET merupakan reset secara hardware yang dapat dilakukan setiap saat.



Gambar 4-11

## RANGKAIAN RESET TMS32010

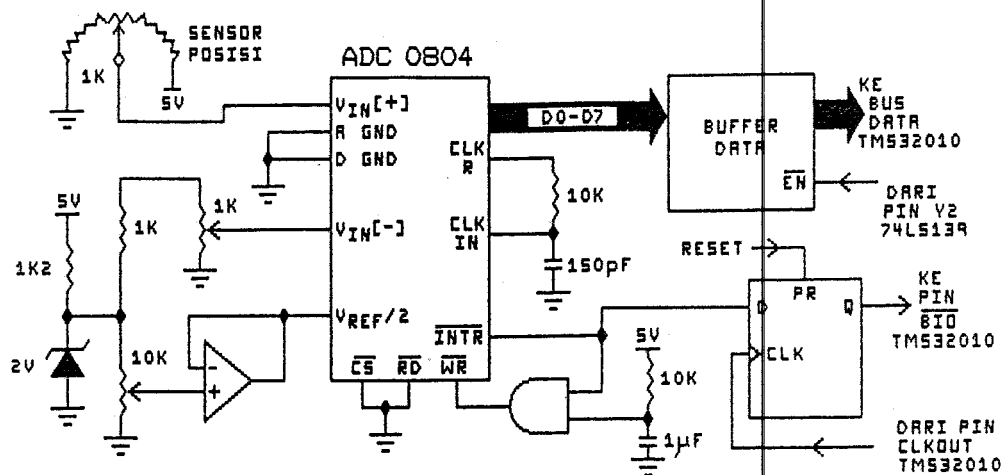
## IV.5 RANGKAIAN I/O

## IV.5.1 Transduser dan Konverter Analog ke Digital

Transduser berfungsi untuk mengubah besaran non-elektrik misalnya besaran mekanik menjadi besaran listrik seperti arus atau tegangan. Pada perencanaan ini untuk mendeteksi posisi sudut putar motor digunakan transduser *potensio putar* yang mengubah besarnya sudut putar motor menjadi tegangan. Tegangan ini diumpankan ke ADC untuk dikonversikan menjadi data biner yang selanjutnya diproses oleh prosesor.

Komponen ADC yang digunakan adalah jenis ADC 0804 dengan resolusi 8 bit dan waktu konversi 100  $\mu$ s. Rangkaiannya ditunjukkan pada gambar 4-12.

ADC 0804 memerlukan frekuensi clock 640 kHz. Ini didapat dengan memberikan komponen luar berupa R dan C yang dihubungkan seri sehingga diperoleh frekuensi yang diperlukan.



Gambar 4-12

## RANGKAIAN TRANSDUSER dan ADC

Kedelapan pin output ADC mempunyai sifat 'tri-state' sehingga dapat langsung dihubungkan ke bus data prosesor, tetapi pada perencanaan ini dipasang buffer untuk menjamin agar data yang dibaca oleh prosesor benar-benar merupakan data yang berlaku. Tegangan referensi yang dibutuhkan oleh ADC 0804 sebesar 2,5 volt dc didapat dengan memasang dioda zener 2,5 volt pada pin 9 ADC 0804. Dioda ini memperoleh arus dari sumber tegangan 5 volt melalui resistor seri 1,2k $\Omega$ .

Untuk memulai proses konversi pada ADC 0804 dilakukan dengan mengaktifkan kaki  $\overline{WR}$  dan  $\overline{CS}$ . Untuk memastikan bahwa ADC sudah selesai melakukan proses konversi dapat diketahui dari pin  $\overline{INTR}$  yang mengeluarkan strobe aktif low. Data biner hasil konversi dapat dikeluarkan dengan mengaktifkan pin  $\overline{RD}$  (sinyal aktif low. Selama input  $\overline{RD}$  high maka bus data ADC dalam

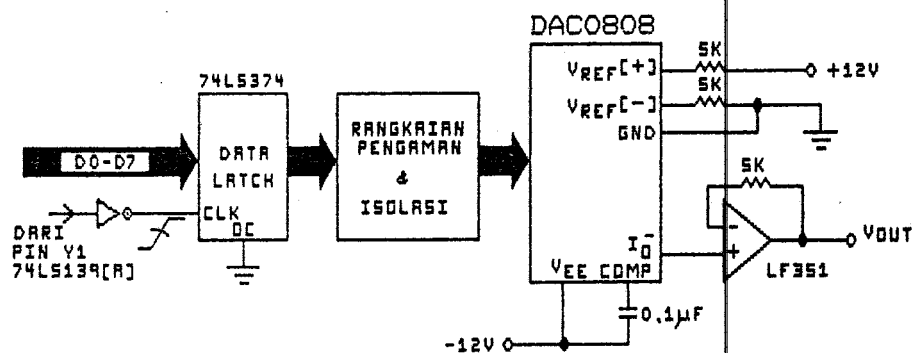
kondisi tri-state.

Pada perencanaan sistem ini ADC diset agar dapat melakukan konversi terus menerus tanpa dikontrol oleh prosesor. Prosesor hanya mengambil data hasil konversi ADC 0804 setiap kali ADC selesai melakukan konversi. Untuk itu maka output  $\overline{\text{INTR}}$  dihubungkan ke pin  $\overline{\text{WR}}$  melalui gate AND seperti tampak pada gambar 4-12, sementara pada input yang satunya terdapat RC seri yang berfungsi untuk mengaktifkan pin  $\overline{\text{WR}}$  sesaat pada waktu permulaan sistem dinyalakan. Sedangkan untuk selanjutnya pin  $\overline{\text{WR}}$  diaktifkan oleh output  $\overline{\text{INTR}}$ . Disamping itu output  $\overline{\text{INTR}}$  juga digunakan untuk meng-interrupt TMS32010 melalui flip-flop 74LS74 sebagai rangkaian sinkronisasi. Kaki  $\overline{\text{CS}}$  dan  $\overline{\text{RD}}$  di-ground. Dengan cara demikian maka setiap kali ADC selesai mengkonversi sinyal analog, ia langsung memberitahu kepada prosesor bahwa data hasil konversi dapat segera diambil. Selanjutnya ADC langsung memulai konversi sinyal yang baru. Pada saat prosesor menerima sinyal interrupt ia segera mengaktifkan buffer melalui address I/O yang telah ditentukan untuk mengambil data hasil konversi.

#### IV.5.2 Konverter Digital ke Analog

Data hasil proses dari prosesor yang berupa besaran biner merupakan data yang akan digunakan untuk mengaktifkan aktuator. Tetapi data hasil ini tidak dapat diumpankan langsung ke aktuator karena aktuator

membutuhkan sinyal besaran analog. Maka data tersebut sebelumnya harus diubah ke besaran analog dengan DAC. DAC yang digunakan adalah tipe 0808 dengan resolusi 8 bit dan memiliki waktu konversi 150 ns. Rangkaianya seperti pada gambar 4-13.



Gambar 4-13

#### RANGKAIAN LATCH dan DAC

DAC ini diberi tegangan referensi sebesar 12 volt agar memberikan sinyal output maksimum 12 volt. Input -input digital pada DAC ini tidak bersifat "tri-state", jadi tidak dapat dihubungkan langsung ke bus data prosesor. Maka diperlukan sebuah data latch untuk menghindari penerimaan data yang salah. Sebagai data latch digunakan tipe 74LS374 merupakan data latch 8 bit. Pin Clock(11) yang aktif pada positive-edge dihubungkan dengan output Y1 dari dekoder 74LS139(a) melalui sebuah inverter. Jadi bila prosesor mengaktifkan sinyal  $\overline{WE}$  dan address I/O menunjuk pada address 1 berarti prosesor akan mengaktifkan 74LS374 dan data diteruskan ke DAC.

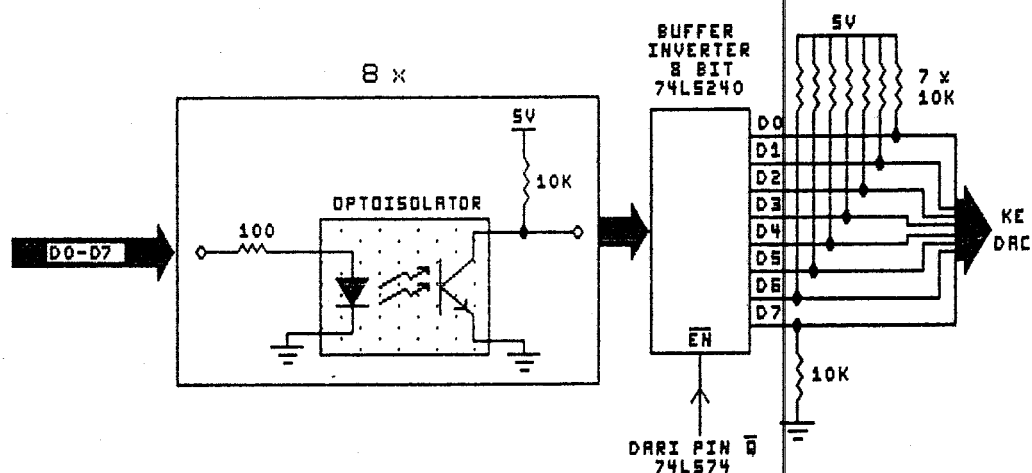


Sedangkan bila pin clock berada pada level "0" atau "1" atau pada negatif-edge maka data yang lama dipertahankan.

#### IV.5.3 Rangkaian Pengaman dan Isolasi

Untuk mengamankan prosesor serta komponen-komponen lainnya dari adanya loncatan-loncatan arus yang kemungkinan ditimbulkan oleh kontak-kontak pada komutator motor atau untuk mencegah masuknya arus besar yang bisa terjadi bila transistor daya mengalami kerusakan, maka dipasang komponen pengaman yang sekaligus sebagai isolasi antara sistem kontroler dengan rangkaian output yaitu menggunakan 'opto-isolator'. Di samping itu jika sistem kontroler berada pada kondisi reset, motor tidak boleh berputar, karena motor yang berputar pada keadaan ini tidak terkontrol sehingga mengakibatkan motor akan berputar terus jika tegangan supply-nya tidak dimatikan. Hal ini akan mengakibatkan rusaknya sensor potensio. Untuk menghindari hal ini, maka input enable buffer inverter 74LS240 harus disable pada waktu sistem dalam keadaan reset. Pin ini dihubungkan dengan output  $\bar{Q}$  dari 74LS74. Dengan cara ini maka ketika sinyal reset aktif, output buffer inverter menjadi tri-state. Dalam keadaan tri-state ini bus data D0-D6 di-pull-up, sedangkan D7 di-pull-down. Jadi input DAC akan menerima data 01111111 yang berarti akan memberikan tegangan output sekitar setengah tegangan referensi. Tegangan output tersebut kemudian diumpankan ke penguat dif-

ferensial sehingga menghasilkan output 0 volt. Dengan demikian bila sinyal reset diaktifkan, maka motor langsung berhenti berputar sampai sinyal reset dilepaskan lagi. Rangkaianya seperti tampak pada gambar 4-14.



Gambar 4-14

#### RANGKAIAN PENGAMAN dan ISOLASI

Harga resistor pembatas arus pada dioda optoisolator dapat ditentukan sebagai berikut:

- tegangan output level high dari 74LS374 sekitar 4 volt,
- arus maksimum yang diperbolehkan pada dioda opto sekitar 20 mA, dan forward bias yang dibutuhkan untuk membuat dioda ON adalah 2 volt.

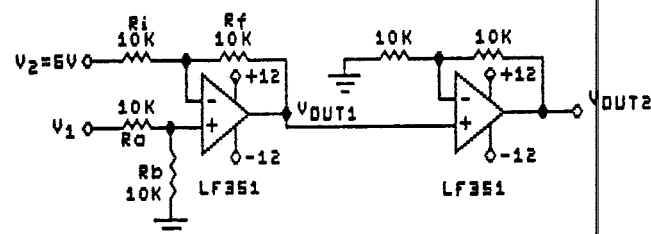
Jadi harga resistor pembatas arus adalah:

$$\begin{aligned}
 R &= \frac{V_Q - 2}{20 \text{ mA}} \\
 &= \frac{(4 - 2)}{20 \text{ mA}} = 100 \, \Omega
 \end{aligned}$$

## IV.6 RANGKAIAN AMPLIFIER

### IV.6.1 Differensial Amplifier

Tegangan output yang dihasilkan oleh DAC mempunyai jangkauan antara 0 volt hingga  $V_{ref}=12$  volt. Tegangan ini tidak dapat langsung digunakan untuk menggerakkan motor secara bolak-balik. Untuk itu maka output ini harus diubah sehingga mempunyai jangkauan tegangan antara -6 volt hingga +6 volt. Dengan menggunakan differensial amplifier hal ini dapat dilakukan. Pada gambar 4-15 ditunjukkan rangkaian differensial amplifier.



Gambar 4-15

### RANGKAIAN DIFFERENSIAL AMPLIFIER

Pada rangkaian differensial amplifier di atas  $V_{out1}$  dapat dicari dengan menggunakan teorema superposisi sebagai berikut:

$$V_{out1} = V_{out1} \Big|_{V_2=0} + V_{out1} \Big|_{V_1=0}$$

Untuk  $V_2 = 0$ ,

$$V_{out1} = \left[ \frac{R_f}{R_i} + 1 \right] \left[ \frac{R_b}{R_a + R_b} \right] \times V_1$$

Untuk  $V_1 = 0$ ,

$$V_{out1} = - \frac{R_f}{R_i} \times V_2$$

Maka

$$V_{out1} = \left[ \frac{R_f}{R_i} + 1 \right] \left[ \frac{R_b}{R_a + R_b} \right] \times V_1 - \frac{R_f}{R_i} \times V_2$$

Dengan memilih  $R_f = R_i = R_a = R_b$  didapat:

$$V_{out1} = V_1 - V_2 \quad (4.1)$$

$V_1$  adalah tegangan input yang berasal dari output DAC sedangkan  $V_2$  adalah tegangan konstan 6V.

Jadi:

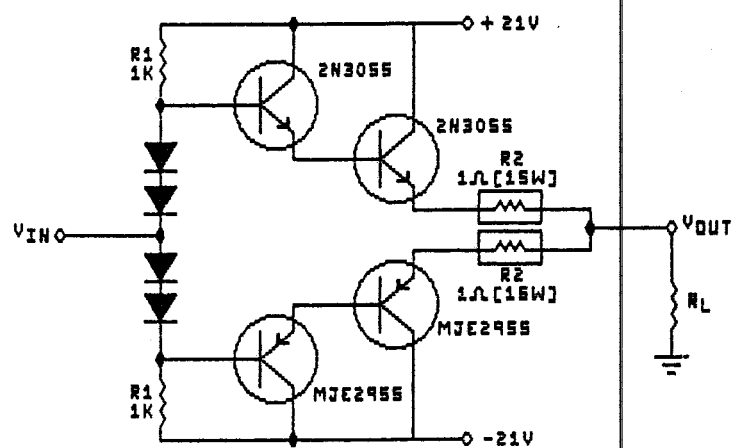
$$V_{out1} = V_1 - 6 \quad (4.2)$$

Dari persamaan (4.2) dapat diketahui bahwa bila  $V_1$  berada dalam jangkauan -12 hingga +12 volt maka jangkauan  $V_{out1}$  adalah -6 hingga +6 volt. Output ini kemudian dikuatkan lagi oleh non-inverting amplifier dengan penguatan tegangan = 2, sehingga  $V_{out2}$  mempunyai jangkauan antara -12 hingga +12 volt. Akan tetapi mengingat bahwa tegangan supply yang digunakan pada OP-AMP di atas adalah  $\pm 12$  volt, output maksimum tidak bisa mencapai  $\pm 12$  volt karena akan mengalami saturasi pada tegangan output sekitar  $\pm 11$  volt. Tetapi hal ini tidak mempengaruhi kerja sistem karena tegangan output  $\pm 11$  volt sudah cukup untuk menggerakkan motor.

#### IV.6.2 Power Amplifier

Rangkaian penguat daya yang dipakai pada perencanaan ini merupakan penguat *simetris komplementer kelas B*. Penguat ini menggunakan transistor tipe NPN dan PNP yang dihubungkan secara komplementer simetris supaya dapat menggerakkan motor dalam dua arah yang berlawanan. Masing-masing transistor dirangkai dengan konfigurasi common-collector sehingga mempunyai penguatan arus  $A_I$  besar dan penguatan tegangan  $A_V \approx 1$ .

Untuk memperoleh penguatan arus yang lebih besar maka masing-masing transistor dihubungkan darlington. Rangkaian penguat daya ditunjukkan pada gambar 4-16.



Gambar 4-16

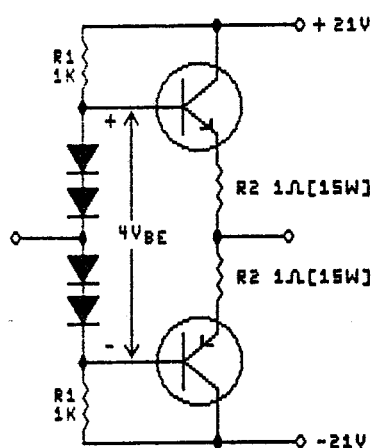
#### RANGKAIAN POWER AMPLIFIER KELAS B

##### IV.6.2.1 Perencanaan Titik Kerja DC

Rangkaian pada gambar 4-16 dapat disederhanakan seperti gambar 4-17 berikut. Hubungan Darlington dari

masing-masing transistor diganti dengan satu transistor dimana harga  $\beta_{dc}$  yang baru adalah hasil kali dari  $\beta_{dc}$  tiap-tiap transistor.

Dalam perencanaan ini diasumsikan bahwa transistor NPN dan PNP yang dipakai benar-benar komplementer, artinya kurva transkonduktansi keduanya adalah serupa; dan mempunyai harga  $\beta_{dc}$  yang sama. Untuk keperluan itu maka digunakan tipe 2N3055(NPN) dan MJE2955(PNP) yang keduanya adalah komplementer.



Gambar 4-17

#### RANGKAIAN EKIVALEN DC

Tegangan supply yang diberikan pada rangkaian penguat ini merupakan supply simetris sebesar  $\pm 21$  volt. Arus kolektor dan emiter mengalir ke bawah melalui transistor NPN dan PNP. Karena hubungan seri, maka  $I_C$  seri dari transistor adalah sama, demikian juga  $I_E$  seri juga sama. Maka dari sini dapat dicari  $V_{CEQ}$  sebagai berikut:

$$\begin{aligned}
 V_{CEQ} &= \frac{+V_{CC} - V_{EE} - 2 I_E R_2}{2} & (4.3) \\
 &= \frac{+21 - (-21) - 2 I_E R_2}{2} ; R_2 = 1 \Omega \\
 &= 21 - I_E \text{ volt.}
 \end{aligned}$$

Pada rangkaian penguat di atas digunakan bias dioda, cara yang umum dipakai untuk mem-bias penguat balance kelas B. Transistor tersebut benar-benar komplementer dan kurva dioda kompensasi sesuai dengan kurva transkonduktansi dari transistor. Karena itu setengah bagian atas dari rangkaian adalah sebuah cermin NPN dan setengah bagian bawah adalah cermin PNP. Gabungan dari keduanya disebut cermin komplementer. Berdasarkan konsep cermin arus, arus kolektor adalah pantulan dari arus yang melalui resistor bias  $R_1$ . Maka  $I_C$  dapat dicari:

$$I_C = \frac{+V_{CC} - V_{EE} - 4 V_{BE}}{2 R_1} \quad (4.4)$$

$R_1$  dipilih 1 K $\Omega$ , maka:

$$I_{CQ} = I_C = \frac{42 - 4 \times 0,7}{2 \times 1 \text{ K}} = 19,6 \text{ mA}$$

Harga  $\beta_{dc}$  transistor 2N3055 dan MJE2955 dari lembar data besarnya sekitar 20. Maka  $\beta_{dc}$  yang baru untuk transistor adalah  $\beta_{dc} = 20 \times 20 = 400$ . Dari sini  $I_E$  dapat dihitung:

$$\begin{aligned}
 I_E = I_C + I_B &= I_C(1 + 1/\beta_{dc}) & (4.5) \\
 &= 19,6(1 + 1/400) \text{ mA}
 \end{aligned}$$

$$I_E = 19,649 \approx 19,65 \text{ mA.}$$

Oleh karena  $I_E \ll 21$ , maka  $V_{CEQ} \approx 21 \text{ volt.}$

#### IV.6.2.2 Analisa AC

##### - Impedansi Input

Impedansi input sinyal besar dari transistor adalah:

$$z_{in(basis)} = \beta_{dc}(R_L + R_e') \quad (4.6)$$

dengan  $R_L$  = resistansi beban.

$R_e'$  = resistansi ac dioda emiter.

Dari hasil pengukuran didapat resistansi beban (motor)  $R_L = 10 \Omega$ . Sedangkan  $R_e'$  diperoleh melalui rumus berikut:

$$\begin{aligned} R_e' &= 25 \text{ mV} / I_E \\ &= 25 \text{ mV} / 19,65 \text{ mA} \\ &= 1,27 \Omega. \end{aligned} \quad (4.7)$$

$$\begin{aligned} \text{Maka: } z_{in(basis)} &= 400(10 + 1,27) = 4508 \\ &= 4,5 \text{ K}\Omega \end{aligned}$$

##### - Penguatan tegangan

$$\begin{aligned} A_v &= \frac{R_L}{R_L + R_e'} \\ &= \frac{10}{10 + 1,27} \\ &= 0,88 \end{aligned} \quad (4.8)$$

##### - Penguatan Daya

$$\begin{aligned} A_p &= \beta_{dc} \frac{R_L}{R_L + R_e'} \\ &= 400 \times 0,88 \\ &= 352 \end{aligned} \quad (4.9)$$



- Daya Output Maksimum

$$P_{o(max)} = \frac{V_{CEQ} I_{C(sat)}}{2} \quad (4.10)$$

Sedangkan:  $I_{C(sat)} = V_{CEQ}/R_L$

$$\begin{aligned} \text{Jadi: } P_{o(max)} &= \frac{V_{CEQ}^2}{2 R_L} = \frac{21^2}{2 \times 10} \\ &= 22,05 \text{ watt.} \end{aligned}$$

Dari lembar data, daya output maksimum yang diperbolehkan untuk transistor 2N3055 dan MJE2955 adalah 40 watt, jadi daya output dari rangkaian di atas masih berada dalam jangkauan yang diperbolehkan dan memenuhi syarat untuk digunakan.

- Efisiensi Daya Maksimum

Efisiensi daya adalah rasio antara daya output dengan daya input dc, atau  $\eta = P_o/P_{DC}$ .

Disipasi daya untuk rangkaian penguat kelas B di atas adalah:

$$P_{DC} = (V_{CC} - V_{EE}) I_{DC} \quad (4.11)$$

dengan  $I_{DC}$  adalah arus pada transistor yang dirata-rata dalam satu cycle. Karena arus yang mengalir pada transistor merupakan sinyal setengah gelombang untuk masing-masing bagian (bagian NPN dan bagian PNP), maka:

$$\begin{aligned} I_{DC} &= \frac{I_{C(sat)}}{\pi} \\ &= \frac{V_{CEQ}}{\pi \times R_L} \end{aligned} \quad (4.12)$$

$$= \frac{21}{\pi \times 10}$$

$$= 0,67 \text{ A.}$$

Persamaan (4.11) didapat:

$$P_{DC} = 42 \text{ V} \times 0,67 \text{ A} = 28,14$$

Jadi:

$$n = \frac{22,05}{28,14} \approx 0,7836$$

$$= 78,36 \%$$

### III.7 MOTOR

Motor yang dipergunakan adalah motor DC dengan medan magnet tetap. Seperti telah dijelaskan bahwa pengaturan kecepatan motor dapat dilakukan dengan mengubah-ubah tegangan jangkar yang berarti mengatur tegangan sumber yang dikenakan pada motor. Tegangan yang dibutuhkan motor adalah  $\pm 12$  Volt dengan arus  $\pm 1$  ampere, cukup kuat untuk menggerakkan beban yang tidak terlalu berat karena pada badan motor itu terdapat roda gigi untuk memperbesar torsi motor.

### IV.8 PERENCANAAN PERANGKAT LUNAK (SOFTWARE)

Pada perencanaan perangkat lunak ini ada dua bagian yang akan dibahas yaitu mengenai proses transfer data dari komputer IBM PC/XT ke memori program sistem kontroler dan sekaligus mengecek kebenaran datanya. Kedua adalah mengenai proses pengaturan motor oleh sistem kontroler.

#### IV.8.1 Routine Transfer Data

Pada proses pengiriman data dari komputer IBM PC/XT ke memori program sistem kontroler, prosedur yang dilakukan adalah sebagai berikut:

- reset TMS23010,
- memindahkan data ke segmen yang telah ditentukan,
- mencocokkan data yang telah dikirim dengan data asal,
- melepaskan reset TMS32010.

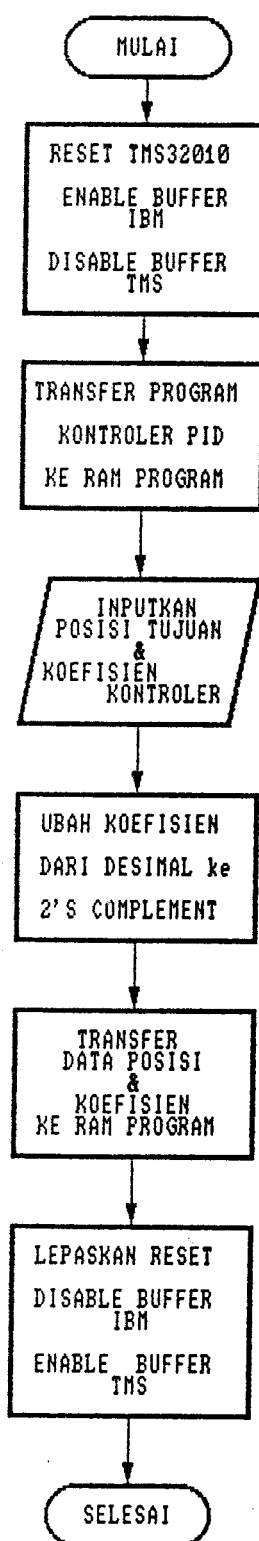
Routine ini ditulis dalam bahasa tingkat tinggi Turbo Pascal versi 5.0, sementara kode-kode instruksi TMS32010 yang akan ditransfer ditulis dalam bentuk tabel bilangan heksadesimal dan termasuk di dalam bagian program tersebut.

Harga koefisien-koefisien dan harga posisi yang dikehendaki di-input-kan melalui keyboard.

Diagram alir dari prosedur di atas ditunjukkan pada gambar 4-18.

#### IV.8.2 Routine Pengaturan Motor oleh TMS32010

Dalam melakukan pengaturan ini prosesor mengerjakan tiga langkah yaitu menentukan posisi yang dikehendaki, membaca posisi sudut putar motor pada saat itu yang selanjutnya diproses dengan menggunakan rumus kontroler PID seperti telah dijelaskan pada bab II. Hasil proses ini kemudian dibandingkan dengan posisi yang telah ditentukan. Deviasi dari perbandingan ter-



Gambar 4-18

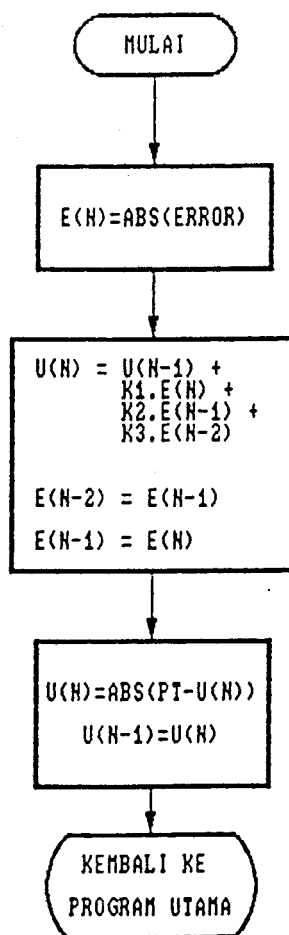
DIAGRAM ALIR PROGRAM TRANSFER DATA

sebut digunakan untuk menggerakkan motor ke posisi yang diinginkan.

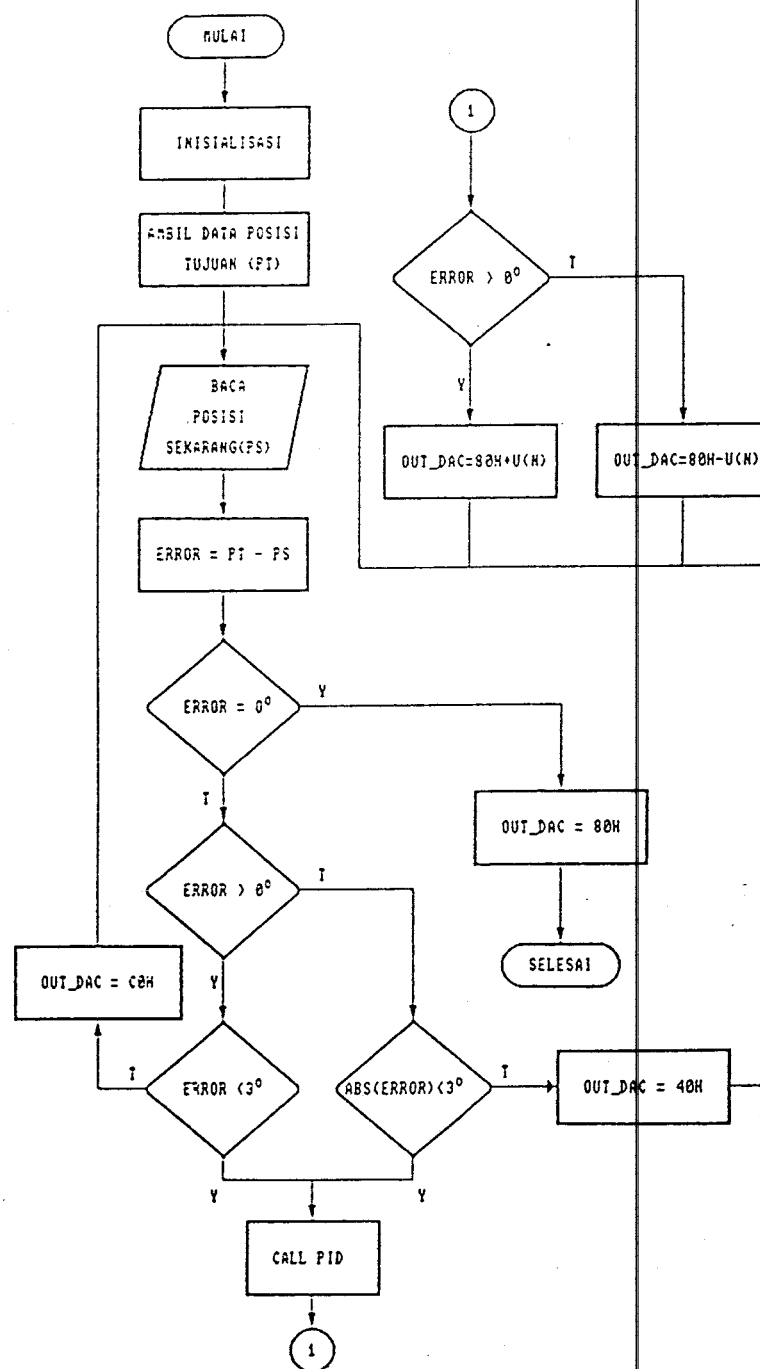
Prosesor TMS32010 melakukan proses looping untuk setiap saat mencek posisi sudut putar motor pada saat itu. Bila motor telah mencapai posisi yang dikehendaki maka motor berhenti berputar.

Diagram alir pengaturan motor ditunjukkan pada gambar 4-19.

#### SUBROUTINE PID



## PROGRAM UTAMA



Gambar 4-19

DIAGRAM ALIR PROGRAM PENGATURAN MOTOR

## **BAB V**

### **PENGUJIAN ALAT**

#### **V.1 PENDAHULUAN**

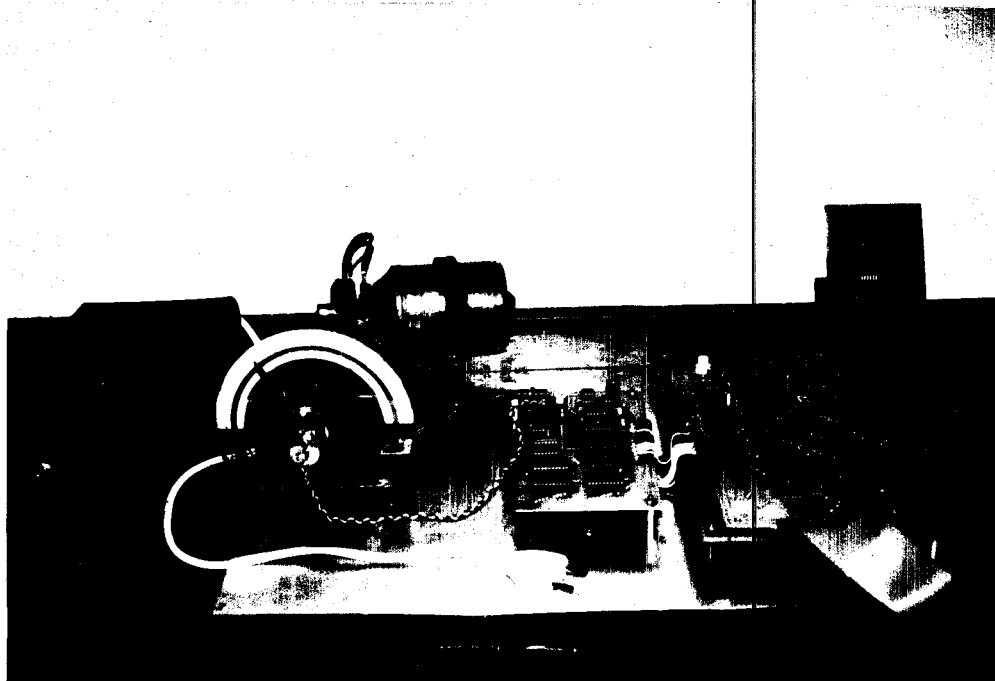
Pada bab ini akan diberikan hasil pengujian alat yang telah dibuat berdasarkan perencanaan pada Bab IV. Pengujian dibagi menjadi tiga bagian berdasarkan blok-blok yang diukur, yaitu pengujian pada:

1. Modul Kontroler, mengenai pengukuran pada Prosesor TMS32010 yang meliputi clock output (CLKOUT), strobe  $\overline{\text{MEN}}$ ,  $\overline{\text{WE}}$  dan  $\overline{\text{DEN}}$ .
2. Modul Sensor, meliputi pengukuran clock ADC 0804, sinyal output interrupt INTR dan hasil konversi posisi derajat ke data biner.
3. Modul Aktuator, meliputi pengukuran hasil konversi data biner ke tegangan analog, output Differensial Amplifier, output Power Amplifier dan ketelitian posisi derajat.

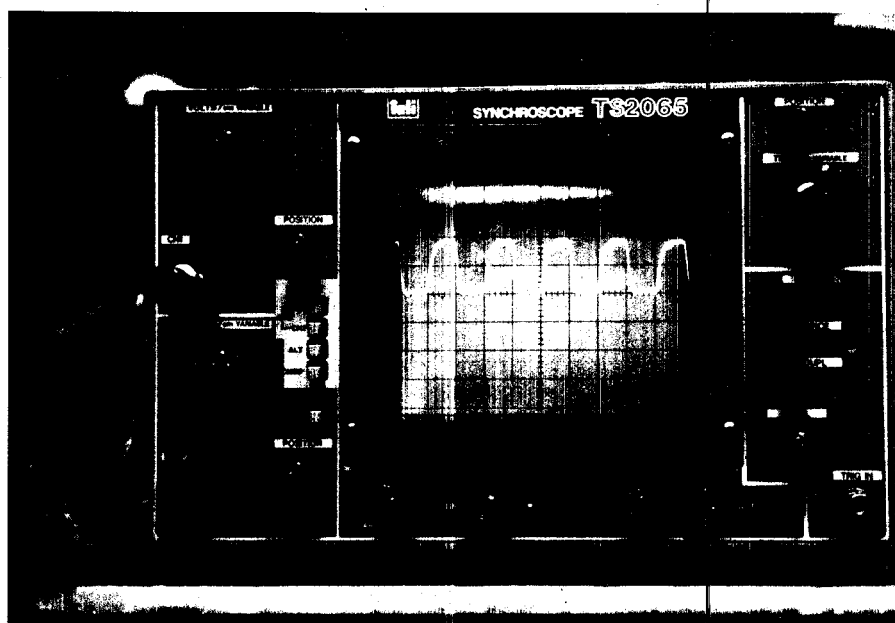
Adapun bentuk fisik alat tersebut ditunjukkan pada gambar 5-1.

#### **V.2 PENGUJIAN MODUL KONTROLER**

Prosesor TMS32010 bekerja dengan clock internal dengan frekuensi kristal 10 MHz. Sesuai dengan prinsip kerja clock internal pada TMS32010, frekuensi output pada pin CLKOUT adalah 1/4 frekuensi kristal, jadi  $f_C = 2,5 \text{ MHz}$ . Hasil pengukuran dapat dilihat pada gambar 5-2.



Gambar 5-1  
BENTUK FISIK ALAT YANG DIBUAT



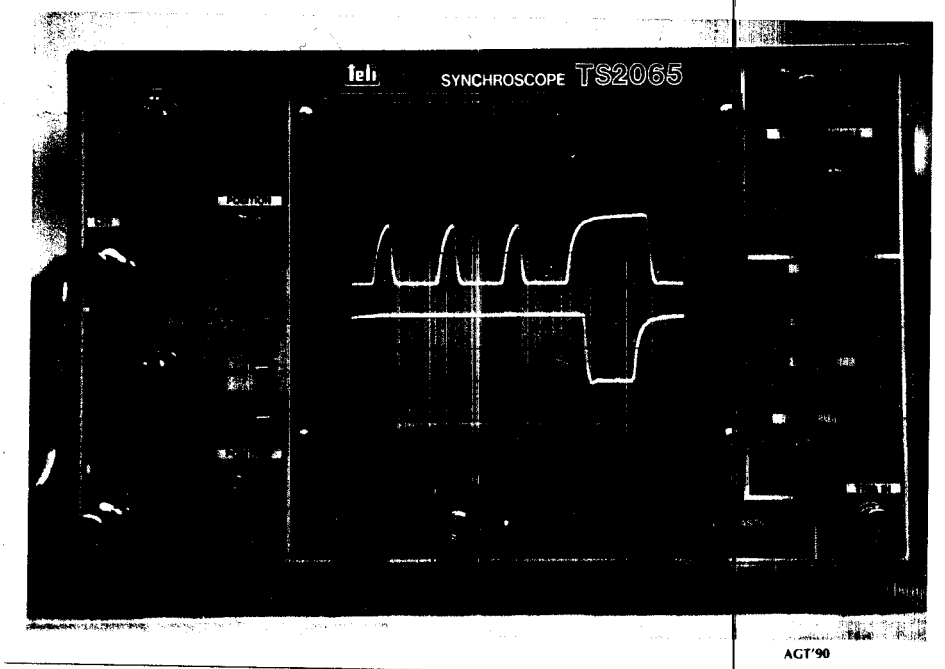
AGT'90

Gambar 5-2  
FREKUENSI CLKOUT TMS32010



Strobe  $\overline{\text{MEN}}$  selalu aktif low pada setiap machine cycle kecuali bila  $\overline{\text{WE}}$  atau  $\overline{\text{DEN}}$  aktif. Jadi  $\overline{\text{MEN}}$  aktif setiap melakukan fetch instruksi (lihat gambar 5-3 dan gambar 5-4).

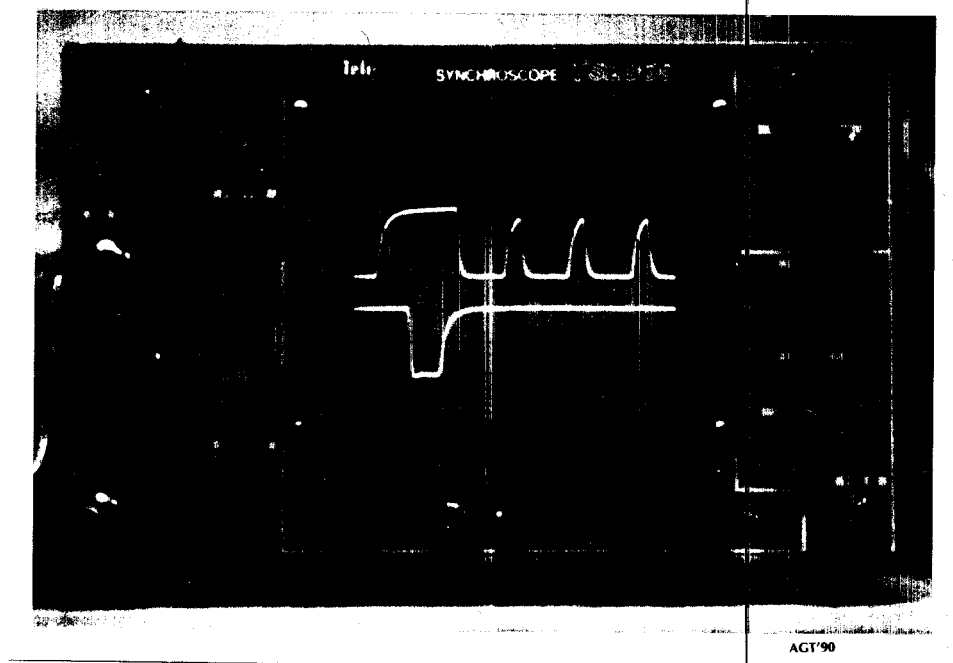
Strobe  $\overline{\text{DEN}}$  hanya aktif selama cycle pertama dari instruksi IN seperti ditunjukkan pada gambar 3-11. Hasil pengukuran ditunjukkan pada gambar 5-3. Pada gambar tersebut tampak bahwa bila  $\overline{\text{DEN}}$  aktif, maka  $\overline{\text{MEN}}$  menjadi tidak aktif (high).



Gambar 5-3  
STROBE  $\overline{\text{MEN}}$  (atas) dan  $\overline{\text{DEN}}$  (bawah)  
UNTUK INSTRUKSI IN

Strobe  $\overline{\text{WE}}$  hanya aktif selama cycle pertama dari instruksi OUT atau cycle kedua dari instruksi TBLW

seperti ditunjukkan pada gambar 3-12 dan 3-14. Oleh karena instruksi TBLW tidak digunakan di sini, maka pengukuran strobe  $\overline{WE}$  hanya dilakukan untuk instruksi OUT seperti ditunjukkan pada gambar 5-4.



Gambar 5-4  
STROBE  $\overline{MEN}$  (atas) dan  $\overline{WE}$  (bawah)  
UNTUK INSTRUKSI OUT

### V.3 PENGUJIAN MODUL SENSOR

#### V.3.1 Pengujian ADC 0804

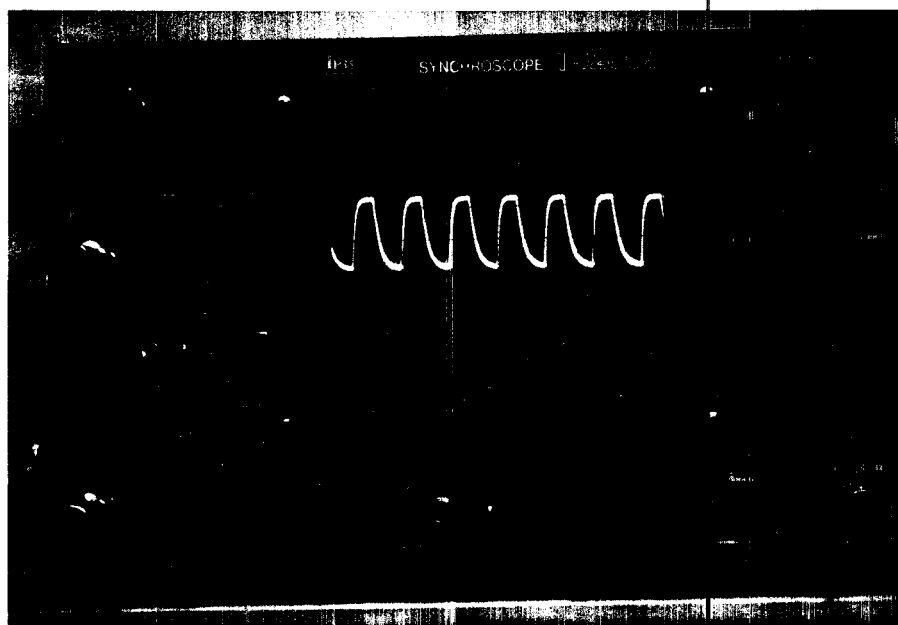
Clock ADC 0804 pada alat yang dibuat ini menggunakan clock internal. Frekuensi clock diperoleh dengan rumus:

$$f_{CLK} \approx \frac{1}{1,1 RC}$$

Untuk harga  $R = 10k$  dan  $C = 150pF$  maka:

$$f_{CLK} \approx 606,060 \text{ kHz.}$$

Bentuk clock ditunjukkan pada gambar 5-5.



AGT'90

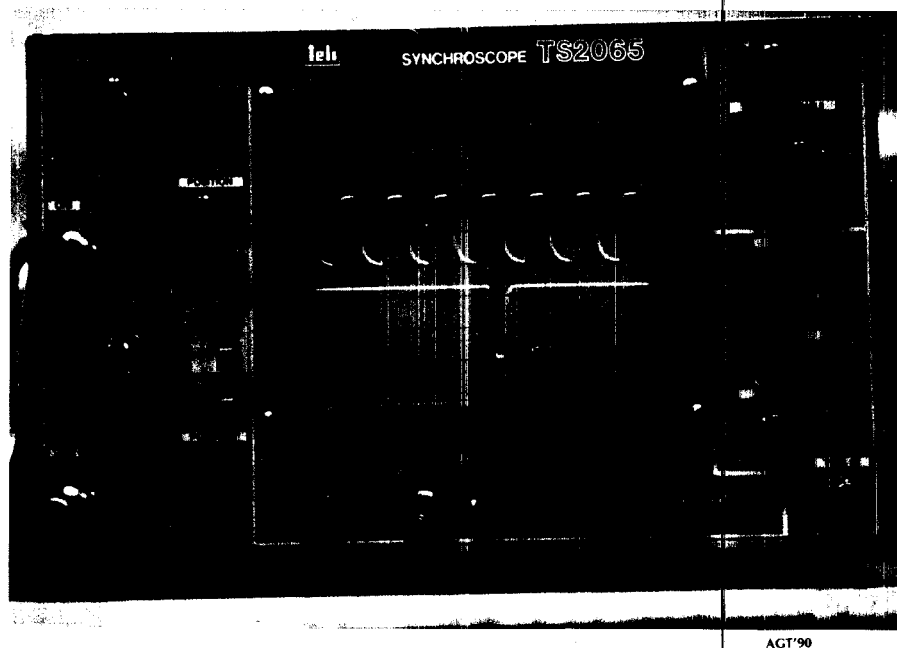
Gambar 5-5

#### CLOCK PADA ADC 0804

Sinyal INTR pada ADC 0804 selalu aktif bila ADC telah selesai melakukan proses konversi. Sinyal aktif INTR juga menandakan bahwa ADC telah bekerja. Bentuk pulsa sinyal INTR ditunjukkan pada gambar 5-6.

#### V.3.2 Konversi Posisi Derajat ke Data Byte

Transduser potensiometer yang dikopel dengan poros motor mengkonversikan posisi sudut putar motor ke tegangan analog. Tegangan ini selanjutnya oleh ADC dikonversi lagi menjadi byte data untuk dikirim ke prosesor. Hasil konversi untuk tiap derajat kemudian



AGT'90

Gambar 5-6

## SINYAL OUTPUT INTR ADC 0804

dicatat dan disimpan dalam bentuk data yang nantinya akan dipakai oleh prosesor sebagai posisi acuan. Data posisi ini berada dalam jangkauan antara  $0^{\circ}$  -  $180^{\circ}$ . Tabel hasil konversi posisi derajat ke byte data adalah sebagai berikut:

Tabel 5-1. KONVERSI POSISI DERAJAT KE DATA BYTE.

Posisi	Byte	Posisi	Byte	Posisi	Byte	Posisi	Byte
$0^{\circ}$	02H	$50^{\circ}$	45H	$100^{\circ}$	91H	$150^{\circ}$	D6H
$5^{\circ}$	08H	$55^{\circ}$	4CH	$105^{\circ}$	98H	$155^{\circ}$	DCH
$10^{\circ}$	0FH	$60^{\circ}$	54H	$110^{\circ}$	9FH	$160^{\circ}$	E2H
$15^{\circ}$	16H	$65^{\circ}$	5BH	$115^{\circ}$	A5H	$165^{\circ}$	E9H
$20^{\circ}$	1CH	$70^{\circ}$	62H	$120^{\circ}$	ADH	$170^{\circ}$	EFH
$25^{\circ}$	25H	$75^{\circ}$	6AH	$125^{\circ}$	B4H	$175^{\circ}$	F5H
$30^{\circ}$	2AH	$80^{\circ}$	72H	$130^{\circ}$	BBH	$180^{\circ}$	FCH
$35^{\circ}$	31H	$85^{\circ}$	79H	$135^{\circ}$	C1H		
$40^{\circ}$	37H	$90^{\circ}$	82H	$140^{\circ}$	C9H		
$45^{\circ}$	3EH	$95^{\circ}$	89H	$145^{\circ}$	CFH		

## V.4 MODUL AKTUATOR

### V.4.1 Pengujian DAC 0808

DAC 0808 pada alat yang dibuat menggunakan tegangan referensi +12 volt. Tegangan output DAC dapat dihitung dari rumus:

$$V_o = V_{ref} \left( \frac{A_1}{2} + \frac{A_2}{4} + \dots + \frac{A_8}{256} \right)$$

$$= 12V \left( \frac{A_1}{2} + \frac{A_2}{4} + \dots + \frac{A_8}{256} \right)$$

dengan  $A_1$  MSB dan  $A_8$  LSB.

Hasil pengukuran untuk beberapa sampel input adalah:

Tabel 5-2. HASIL KONVERSI DAC

Input Byte Data	$V_o$ Perhitungan (V)	$V_o$ Pengukuran (V)
00	0,00	0,006
10	0,75	0,748
20	1,50	1,499
30	2,25	2,240
40	3,00	2,990
50	3,75	3,730
60	4,50	4,480
70	5,25	5,220
80	6,00	5,980
90	6,75	6,720
A0	7,50	7,470
B0	8,25	8,210
C0	9,00	8,960
D0	9,75	9,700
E0	10,50	10,450
F0	11,25	11,060

### V.4.2 Pengujian Differensial Amplifier

Differensial Amplifier berfungsi untuk menurunkan level tegangan output dari DAC yang berada dalam jang-

kauan antara 0 hingga  $V_{maks}$  menjadi  $-1/2V_{maks}$  hingga  $+1/2V_{maks}$ . Ini dimaksudkan agar dapat menggerakkan motor dalam dua arah.

Berdasarkan persamaan (4.2) maka tegangan output differensial amplifier adalah:

$$V_o(\text{Diff}) = V_o(\text{DAC}) - 6$$

Hasil pengukuran untuk beberapa sampel input ditunjukkan pada tabel 5-3.

Tabel 5-3. PENGUKURAN DIFFERENSIAL AMPLIFIER

Input Byte Data	$V_o(\text{DAC})$ (V)	$V_o(\text{Diff})$ Perhitungan (V)	$V_o(\text{Diff})$ Pengukuran (V)
00	0,006	-5,994	-5,840
10	0,748	-5,252	-5,090
20	1,499	-4,501	-4,330
30	2,240	-3,760	-3,590
40	2,990	-3,010	-2,830
50	3,730	-2,270	-2,090
60	4,480	-1,520	-1,330
70	5,220	-0,780	-0,594
80	5,980	-0,020	0,174
90	6,720	0,720	0,918
A0	7,470	1,470	1,673
B0	8,210	2,210	2,410
C0	8,960	2,960	3,170
D0	9,700	3,700	3,910
E0	10,450	4,700	4,660
F0	11,060	5,060	5,270

#### V.4.3 Pengujian Power Amplifier

Output differensial amplifier yang mempunyai level tegangan  $-1/2V_{maks}$  hingga  $+1/2V_{maks}$  kemudian diperkuat oleh non-inverting amplifier sehingga level tegangan output menjadi  $-V_{maks}$  hingga  $+V_{maks}$ . Level tegangan tersebut kemudian diumpankan ke power amplifier. Hasil

pengukuran ditunjukkan pada tabel 5-4.

Tabel 5-4. PENGUKURAN POWER AMPLIFIER

$V_o(\text{Diff})$ (V)	$V_o(\text{non-inv})$ (V)	$V_o(\text{PA})$ (V)
-5,840	-10,470	-10,160
-5,090	-10,190	-9,870
-4,330	-8,680	-8,370
-3,590	-7,180	-6,880
-2,830	-5,670	-5,380
-2,090	-4,180	-3,900
-1,330	-2,670	-2,430
-0,594	-1,181	-1,098
0,174	0,358	0,338
0,918	1,840	1,736
1,673	3,360	3,110
2,410	4,850	4,580
3,170	6,360	6,080
3,910	7,850	7,560
4,660	9,350	9,070
5,270	10,570	10,280

#### V.4.4 Pengujian Terhadap Ketelitian Posisi Derajat

Pada pengujian ini akan dilihat hasil penunjukan posisi derajat yang sebenarnya kemudian dibandingkan dengan posisi yang diinginkan. Dalam hal ini posisi yang dikehendaki di-input-kan melalui keyboard komputer IBM PC/XT sedangkan posisi yang sebenarnya langsung dilihat dari penunjukan jarum pada busur derajat. Hasil pengamatan ditunjukkan pada tabel 5-5.

Tabel 5-5. HASIL PENGAMATAN KETELITIAN POSISI DERAJAT

Posisi Input Keyboard	Hasil Pengamatan
0°	0°
10°	10°
20°	20°

30°	30°
40°	40°
50°	50,5°
60°	60°
70°	70°
80°	80°
90°	90°
100°	100,5°
110°	110°
120°	120°
130°	130°
140°	140°
150°	150°
160°	160°
170°	170°
180°	180°
170°	170°
160°	159,5°
150°	150°
140°	140°
130°	130°
120°	120°
110°	110°
100°	100°
90°	90°
80°	80°
70°	70°
60°	60°
50°	50°
40°	40°
30°	29,5°
20°	20°
10°	10°
0°	0°

Dari hasil-hasil pengukuran dan pengamatan di atas tampak adanya perbedaan antara perhitungan teoritis dengan hasil yang sebenarnya. Hal ini disebabkan oleh toleransi komponen-komponen amplifier yang digunakan seperti resistor-resistor dan OP-AMP. Demikian juga mengenai ketelitian posisi derajat, hal ini sangat dipengaruhi oleh kelinieran potensio putar dan faktor kesalahan konversi oleh ADC.



## **BAB VI**

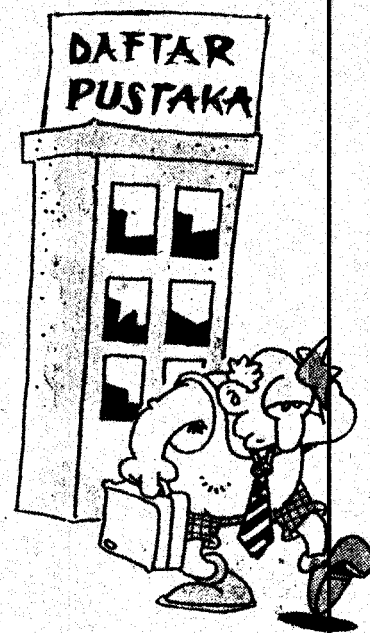
### **KESIMPULAN**

Dari keseluruhan perencanaan dan pembuatan alat dapat diambil beberapa kesimpulan sebagai berikut:

1. Kontroler PID digital merupakan alternatif penggunaan kontroler PID analog dengan beberapa keunggulan yang dimiliki antara lain:
  - parameter-parameternya tidak dipengaruhi oleh faktor luar seperti lama pemakaian alat dan faktor temperatur ruang.
  - mempunyai fleksibilitas tinggi karena koefisien dapat diubah-ubah secara software.
2. Pada pengaturan posisi dengan kontroler PID digital, ketelitian derajat putaran ditentukan oleh resolusi ADC dan kelinieran pada transduser posisi yang dipakai. Dengan menggunakan ADC yang mempunyai resolusi 8 bit, secara ideal akan mampu mendeteksi putaran motor hingga  $256^{\circ}$  bila setiap derajat direpresentasikan dengan 1 bit. Akan tetapi hal ini juga ditentukan oleh kelinieran potensiometer sebagai transduser posisi. Bila digunakan ADC dengan resolusi yang lebih besar misalnya 10 bit, maka akan mampu mendeteksi sebanyak  $1024^{\circ}$ . Namun kembali lagi pada masalah keterbatasan transduser potensiometer yang tidak bisa berputar lebih dari  $360^{\circ}$ . Untuk keperluan tersebut maka rancangan hardware perlu dimodifikasi yaitu transduser potensiometer tidak dikopel

langsung ke poros motor, melainkan melalui roda gigi transmisi dengan perbandingan diameter yang sesuai.

3. Penggunaan Prosesor Sinyal Digital TMS32010 yang memiliki kecepatan eksekusi yang cukup tinggi sebagai pemroses dalam operasi aritmatik untuk mengimplementasikan algoritma kontroler PID digital akan diperoleh output yang real-time.
4. Sistem kontroler ini juga dapat digunakan untuk pengaturan posisi menggunakan motor AC dengan mengubah modul aktuator dan menyesuaikan softwarena. Selain itu alat ini juga bisa diimplementasikan untuk pengaturan kecepatan motor baik DC maupun AC dengan mengubah modul sensor dan/atau aktuator.



## DAFTAR PUSTAKA

1. Borland, Turbo Pascal Ver 4.00, IBM Version, Borland International Inc., Canada, 1987.
2. Carr, Joseph J., Elements of Electronic Instrumentation and Measurement, Reston Publishing Company, Inc., 1979.
3. Coughlin, Robert F., Frederick F. Driscoll, Operational Amplifiers and Linear Integrated Circuits, Prentice-Hall Inc., New York, 1982.
4. Eggebrecht, Lewis C., Interfacing to IBM Personal Computer, Howard W. Sams & Co., 1987.
5. Gopal, M., Digital Control Engineering, Wiley Eastern Limited, New Delhi, 1988.
6. Hall, Douglas V., Microprocessors and Interfacing, McGraw-Hill Book Co., Singapore, 1986.
7. Kaiser, Joe., Electrical Power, Goodheart-Willcox Company, Inc. (Illinois), 1982.
8. Maloney, Timothy J., Industrial Solid-State Electronics, Prentice-Hall, Inc., 1986.
9. Malvino, Albert Paul., Electronic Principles, McGraw-Hill Inc., 1979.
10. Ogata, Katsuhiko., Modern Control Engineering, New Delhi : Prentice-Hall of India., 1982.
11. Quarmby, David J., Signal Processor Chips, Granada Publishing Ltd., London, 1984.
12. Sumarto, Drs., Mesin Arus Searah, Andi Offset Yogyakarta, 1984.

13. ...., Data Conversion/Acquisition, National Semiconductor Corp., USA, 1984.
14. ...., First-Generation TMS320 User's Guide, Texas Instruments Inc., 1988.

# TMS320 FIRST-GENERATION DIGITAL SIGNAL PROCESSORS

JANUARY 1987 - REVISED MARCH 1988

- 160-ns Instruction Cycle
- 144/256-Word On-Chip Data RAM
- 1.5K/4K-Word On-Chip Program ROM
- 4K-Word On-chip Program EPROM (TMS320E15/E17)
- EPROM Code Protection for Copyright Security
- 4K-Word Total External Memory at Full Speed
- 32-Bit ALU/Accumulator
- 16 × 16-Bit Multiplier with a 32-Bit Product
- 0 to 16-Bit Barrel Shifter
- Eight Input and Eight Output Channels
- Dual-Channel Serial Port (TMS32011/C17/E17)
- 16-Bit Bidirectional Data Bus with 50-Mbps Transfer Rate
- Single 5-V Supply
- Packaging: 40-Pin DIP and 44-Pin PLCC
- Commercial and Military Versions Available
- NMOS Technology:
  - TMS32010 ..... 200-ns cycle time
  - TMS32010-14 ..... 280-ns cycle time
- CMOS Technology:
  - TMS320C10-25 ..... 160-ns cycle time
  - TMS320C10 ..... 200-ns cycle time
  - TMS320C10-14 ..... 280-ns cycle time
  - TMS320C15-25 ..... 160-ns cycle time
  - TMS320C15 ..... 200-ns cycle time
  - TMS320E15 (EPROM) . 200-ns cycle time
  - TMS320C17-25 ..... 160-ns cycle time
  - TMS320C17 ..... 200-ns cycle time
  - TMS320E17 (EPROM) . 200-ns cycle time

TMS32010, TMS320C10  
N PACKAGE  
(TOP VIEW)

A1/PA1	1	40	A2/PA2
A0/PA0	2	39	A3
MC/MP	3	38	A4
RS	4	37	A5
INT	5	36	A6
CLKOUT	6	35	A7
X1	7	34	A8
X2/CLKIN	8	33	MEN
BIO	9	32	DEN
VSS	10	31	WE
D8	11	30	VCC
D9	12	29	A9
D10	13	28	A10
D11	14	27	A11
D12	15	26	D0
D13	16	25	D1
D14	17	24	D2
D15	18	23	D3
D7	19	22	D4
D6	20	21	D5

This data sheet provides complete design documentation for all the first-generation devices of the TMS320 family. This facilitates the selection of the devices best suited for user applications by providing all specifications and special features for each TMS320 member. This data sheet is divided into four major sections: architecture, electrical specifications (NMOS and CMOS), timing diagrams, and mechanical data. In each of these sections, generic information is presented first, followed by specific device information. An index is provided for quick reference to specific information about a device.

PRODUCTION DATA documents contain information current as of publication date. Products conform to specifications per the terms of Texas Instruments standard warranty. Production processing does not necessarily include testing of all parameters.

TEXAS  
INSTRUMENTS

POST OFFICE BOX 1443 • HOUSTON, TEXAS 77001

Copyright © 1987, Texas Instruments Incorporated

## TMS320 FIRST-GENERATION DEVICES

### description

The TMS320 family of 16/32-bit single-chip digital signal processors combines the flexibility of a high-speed controller with the numerical capability of an array processor, thereby offering an inexpensive alternative to multichip bit-slice processors. The highly paralleled architecture and efficient instruction set provide speed and flexibility to produce a MOS microprocessor family capable of executing 6.4 MIPS (million instructions per second). The TMS320 family optimizes speed by implementing functions in hardware that other processors implement through microcode or software. This hardware-intensive approach provides the design engineer with processing power previously unavailable on a single chip.

The TMS320 family consists of two generations of digital signal processors. The first generation contains the TMS32010 and its spinoffs, as described in this data sheet. The TMS32020 and TMS320C25 are the second-generation processors, designed for higher performance. Many features are common among the TMS320 processors. Specific features are added in each processor to provide different cost/performance tradeoffs. Software compatibility is maintained throughout the family to protect the user's investment in architecture. Each processor has software and hardware tools to facilitate rapid design.

### introduction

The TMS32010, the first NMOS digital signal processor in the TMS320 family, was introduced in 1983. Its powerful instruction set, inherent flexibility, high-speed number-crunching capabilities, and innovative architecture have made this high-performance, cost-effective processor the ideal solution to many telecommunications, computer, commercial, industrial, and military applications. Since that time, the TMS320C10, a low-power CMOS version of the industry-standard TMS32010, and other spinoff devices have been added to the first generation of the TMS320 family.

The TMS32010 microprocessor is available in two speed versions: TMS32010 (20 MHz) and TMS32010-14 (14 MHz). These devices are capable of executing a 16 x 16-bit multiply with a 32-bit result in a single instruction cycle. On-chip data RAM of 144 words and on-chip program ROM of 1.5K words are available. Full-speed execution of 4K words of off-chip program memory is also possible. The TMS32010-14 provides a low-cost alternative for DSP applications not requiring the maximum operating frequency of the TMS32010. The device provides a direct EPROM interface for cost-effective system development and modification. Both of these devices are pin-for-pin and object-code compatible with the TMS32010 and its development tools.

The TMS320C10 is object-code and pin-for-pin compatible with the TMS32010. It is processed in CMOS technology, achieving a power dissipation less than one-sixth that of the NMOS device. The lower power dissipation makes the TMS320C10 ideal for power-sensitive applications such as digital telephony and portable products. The TMS320C10-25, a 25-MHz version of the TMS320C10, has a 160-ns instruction cycle time and is well suited for high-performance DSP applications. The TMS320C10 is also available in a 280-ns version, the TMS320C10-14. This device provides a low-cost alternative for DSP applications not requiring the maximum operating frequency of the TMS320C10.

The TMS320C15 and TMS320E15 CMOS devices are object-code and pin-for-pin compatible with the TMS32010 and offer expanded on-chip RAM of 256 words and on-chip program ROM or EPROM of 4K words. These devices allow the capability of upgrading performance and reducing power, board space, and system cost without hardware redesign. The TMS320C15 is also available in a 160-ns version, the TMS320C15-25.

TMS320 FIRST-GENERATION  
DEVICES

## introduction (continued)

The TMS320C17 and TMS320E17 also offer expanded on-chip RAM of 256 words and on-chip program ROM or EPROM of 4K words. These devices provide a dual-channel serial interface, on-chip  $\mu$ -law/A-law companding hardware, and a serial port timer. In addition, a 16-bit coprocessor interface provides a direct communication channel to common 4/8-bit microcomputers (no glue logic required), and minimal logic interface to most common 16/32-bit microprocessors. The devices are object-code compatible with the TMS32010 and processed in CMOS technology. The TMS320C17 is also available in a 160-ns version, the TMS320C17-25.

Table 1 provides an overview of the first generation of TMS320 processors with comparisons of memory, I/O, cycle timing, power, package type, technology, and military support. For specific availability, contact the nearest TI sales office.

TABLE 1. TMS320 FIRST-GENERATION DEVICE OVERVIEW

DEVICE	MEMORY				I/O†			CYCLE TIME (ns)	TYP POWER (mW)	PACKAGE TYPE	
	ON-CHIP			OFF-CHIP EXPANSION						SER	PAR
	RAM	ROM	EPROM								
TMS32010‡ (NMOS)	144	1.5K	—	4K	—	8 x 16	—	200	900	40	—
TMS32010-14 (NMOS)	144	1.5K	—	4K	—	8 x 16	—	280	900	40	—
TMS320C10-25 (CMOS)	144	1.5K	—	4K	—	8 x 16	—	160	200	40	44
TMS320C10‡ (CMOS)	144	1.5K	—	4K	—	8 x 16	—	200	165	40	44
TMS320C10-14 (CMOS)	144	1.5K	—	4K	—	8 x 16	—	280	140	40	44
TMS320C15-25 (CMOS)	256	4K	—	4K	—	8 x 16	—	160	250	40	44
TMS320C15‡ (CMOS)	256	4K	—	4K	—	8 x 16	—	200	225	40	44
TMS320E15‡ (CMOS)	256	—	4K	4K	—	8 x 16	—	200	275	40	—
TMS320C17-25 (CMOS)	256	4K	—	—	2	6 x 16	YES	160	275	40	44
TMS320C17 (CMOS)	256	4K	—	—	2	6 x 16	YES	200	250	40	44
TMS320E17 (CMOS)	256	—	4K	—	2	6 x 16	YES	200	275	40	—

†SER = serial; PAR = parallel; CPX = coprocessor interface.

‡Military version available.

§Military version planned; contact nearest TI sales office for availability.

TEXAS  
INSTRUMENTS

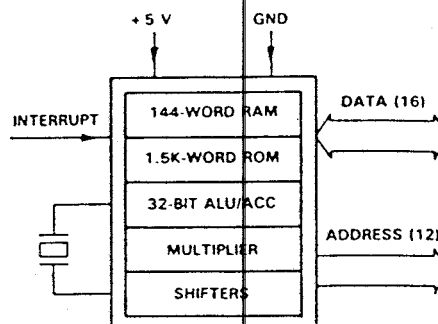
POST OFFICE BOX 1443 • HOUSTON, TEXAS 77001



## TMS320 FIRST-GENERATION DEVICES

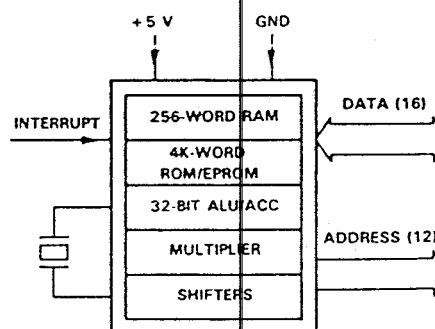
### Key Features: TMS32010/C10

- Instruction Cycle Timing:
  - 160 ns (TMS320C10-25)
  - 200 ns (TMS32010/C10)
  - 280 ns (TMS32010-14/C10-14)
- 144 Words of On-Chip Data RAM
- 1.5K Words of On-Chip Program ROM
- External Memory Expansion up to 4K Words at Full Speed
- 16 x 16-Bit Multiplier with 32-Bit Product
- 0 to 16-Bit Barrel Shifter
- On-Chip Clock Oscillator
- Single 5-V Supply
- Device Packaging:
  - 40-Pin DIP (all devices)
  - 44-Lead PLCC (CMOS only)
- Technology
  - NMOS: TMS32010/10-14
  - CMOS: TMS320C10/C10-25/C10-14



### Key Features: TMS320C15/E15

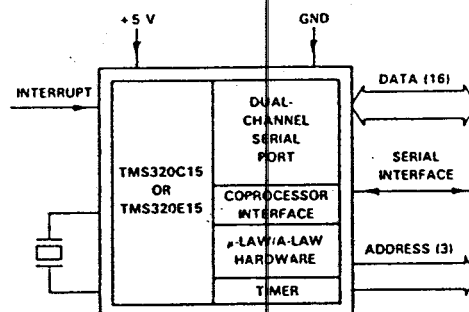
- Instruction Cycle Timing:
  - 160 ns (TMS320C15-25)
  - 200 ns (TMS320C15/E15)
- 256 Words of On-Chip Data RAM
- 4K Words of On-Chip Program ROM (TMS320C15/C15-25)
- 4K Words of On-Chip Program EPROM (TMS320E15)
- EPROM Code Protection for Copyright Security
- External Memory up to 4K Words at Full Speed
- Object-Code and Pin-For-Pin Compatible with TMS32010
- 16 x 16-Bit Multiplier with 32-Bit Product
- 0 to 16-Bit Barrel Shifter
- On-Chip Clock Oscillator
- Single 5-V Supply
- Device Packaging:
  - 40-Pin DIP (all devices)
  - 44-Lead PLCC (TMS320C15/C15-25)
- CMOS Technology



# TMS320 FIRST-GENERATION DEVICES

## Key Features: TMS320C17/E17

- Instruction Cycle Timing:
  - 160 ns (TMS320C17-25)
  - 200 ns (TMS320C17/E17)
- 256 Words of On-Chip Data RAM
- 4K Words of On-Chip Program ROM (TMS320C17/C17-25)
- 4K Words of On-Chip Program EPROM (TMS320E17)
- EPROM Code Protection for Copyright Security
- Object-Code Compatible with TMS32010
- Dual-Channel Serial Port for Full-Duplex Serial Communication
- Serial Port Timer for Standalone Serial Communications
- On-Chip Companding Hardware for  $\mu$ -law/A-law PCM Conversions
- 16-Bit Coprocessor Interface for Common 4/8/16/32-Bit Microcomputers/Microprocessors
- Device Packaging:
  - 40-Pin DIP (all devices)
  - 44-Lead PLCC (TMS320C17/C17-25)
- CMOS Technology



TEXAS  
INSTRUMENTS

POST OFFICE BOX 1443 • HOUSTON, TEXAS 77001

## TMS320 FIRST-GENERATION DEVICES

### architecture

The TMS320 family utilizes a modified Harvard architecture for speed and flexibility. In a strict Harvard architecture, program and data memory lie in two separate spaces, permitting a full overlap of instruction fetch and execution. The TMS320 family's modification of the Harvard architecture allows transfers between program and data spaces, thereby increasing the flexibility of the device. This modification permits coefficients stored in program memory to be read into the RAM, eliminating the need for a separate coefficient ROM. It also makes available immediate instructions and subroutines based on computed values.

### 32-bit ALU/accumulator

The TMS320 first-generation devices contain a 32-bit ALU and accumulator for support of double-precision, two's-complement arithmetic. The ALU is a general-purpose arithmetic unit that operates on 16-bit words taken from the data RAM or derived from immediate instructions. In addition to the usual arithmetic instructions, the ALU can perform Boolean operations, providing the bit manipulation ability required of a high-speed controller. The accumulator stores the output from the ALU and is often an input to the ALU. It operates with a 32-bit wordlength. The accumulator is divided into a high-order word (bits 31 through 16) and a low-order word (bits 15 through 0). Instructions are provided for storing the high- and low-order accumulator words in memory.

### shifters

Two shifters are available for manipulating data. The ALU barrel shifter performs a left-shift of 0 to 16 places on data memory words loaded into the ALU. This shifter extends the high-order bit of the data word and zero-fills the low-order bits for two's-complement arithmetic. The accumulator parallel shifter performs a left-shift of 0, 1, or 4 places on the entire accumulator and places the resulting high-order accumulator bits into data RAM. Both shifters are useful for scaling and bit extraction.

### 16 x 16-bit parallel multiplier

The multiplier performs a 16 x 16-bit two's-complement multiplication with a 32-bit result in a single instruction cycle. The multiplier consists of three units: the T Register, P Register, and multiplier array. The 16-bit T Register temporarily stores the multiplicand; the P Register stores the 32-bit product. Multiplier values either come from the data memory or are derived immediately from the MPYK (multiply immediate) instruction word. The fast on-chip multiplier allows the device to perform fundamental operations such as convolution, correlation, and filtering.

### data and program memory

Since the TMS320 devices use a Harvard architecture, data and program memory reside in two separate spaces. The first-generation devices have 144 or 256 words of on-chip data RAM and 1.5K or 4K words of on-chip program ROM. On-chip program EPROM of 4K words is provided on the TMS320E15/E17. The EPROM cell utilizes standard PROM programmers and is programmed identically to a 64K CMOS EPROM<sup>®</sup> (TMS27C64).

### program memory expansion

The first-generation devices are capable of executing up to 4K words of external memory at full speed for those applications requiring external program memory space. This allows for external RAM-based systems to provide multiple functionality. The TMS320C17/E17 provides no memory expansion capability.

TEXAS  
INSTRUMENTS

POST OFFICE BOX 1441 • HOUSTON, TEXAS 77251

## TMS320 FIRST-GENERATION DEVICES

### microcomputer/microprocessor operating modes (TMS32010/C10/C15/E15)

The TMS32010/C10 and TMS320C15/E15 devices offer two modes of operation defined by the state of the MC/MP pin: the microcomputer mode ( $MC/\overline{MP} = 1$ ) or the microprocessor mode ( $MC/\overline{MP} = 0$ ). In the microcomputer mode, on-chip ROM is mapped into the memory space with up to 4K words of external memory available. In the microprocessor mode, all 4K words of memory are external.

### interrupts and subroutines

The TMS320 first-generation devices contain a four-level hardware stack for saving the contents of the program counter during interrupts and subroutine calls. Instructions are available for saving the device's complete context. PUSH and POP instructions permit a level of nesting restricted only by the amount of available RAM. The interrupts used in these devices are maskable.

### input/output

The 16-bit parallel data bus can be utilized to perform I/O functions in two cycles. The I/O ports are addressed by the three LSBs on the address lines. In addition, a polling input for bit test and jump operations (BIO) and an interrupt pin ( $\overline{INT}$ ) have been incorporated for multitasking.

### serial port (TMS320C17/E17)

Two of the I/O ports on the TMS320C17/E17 are dedicated to the serial port and companding hardware. I/O port 0 is dedicated to control register 0, which controls the serial port, interrupts, and companding hardware. I/O port 1 accesses control register 1, as well as both serial port channels, and the companding hardware. The six remaining I/O ports are available for external parallel interfaces.

The dual-channel serial port is capable of full-duplex serial communication and offers direct interface to combo-codex. Receive and transmit registers that operate with 8-bit data samples are I/O-mapped. Either internal or external framing signals for serial data transfers are selected through the system control register. The serial port clock provides the bit timing for transfers with the serial port, and may be either an input or output. A framing pulse signal provides framing pulses for combo-codec circuits, an 8-kHz sample clock for voice-band systems, or a timer for control applications.

### companding hardware (TMS320C17/E17)

On-chip hardware enables the TMS320C17/E17 to compand (COMpress/exPAND) data in either  $\mu$ -law or A-law format. The companding logic operation is configured via the system control register. Data may be companded in either a serial mode for operation on serial port data (converting between linear and logarithmic PCM) or a parallel mode for computation inside the device. The TMS320C17/E17 allows the hardware companding logic to operate with either sign-magnitude or two's-complement numbers.

### coprocessor port (TMS320C17/E17)

The coprocessor port on the TMS320C17/E17 provides a direct connection to most 4/8-bit microcomputers and 16/32-bit microprocessors. The port is accessed through I/O port 5 using IN and OUT instructions. The coprocessor interface allows the device to act as a peripheral (slave) microcomputer to a microprocessor, or as a master to a peripheral microcomputer. In the microcomputer mode, the 16 data lines are used for the 6 parallel 16-bit I/O ports. In the coprocessor mode, the 16-bit parallel port is reconfigured to operate as a 16-bit latched bus interface. For peripheral transfer, an 8-bit or 16-bit length of the coprocessor port can be selected.

TEXAS  
INSTRUMENTS

POST OFFICE BOX 1443 • HOUSTON, TEXAS 77001

## TMS320 FIRST-GENERATION DEVICES

### instruction set

A comprehensive instruction set supports both numeric-intensive operations, such as signal processing, and general-purpose operations, such as high-speed control. All of the first-generation devices are object-code compatible and use the same 60 instructions. The instruction set consists primarily of single-cycle single-word instructions, permitting execution rates of more than six million instructions per second. Only infrequently used branch and I/O instructions are multicycle. Instructions that shift data as part of an arithmetic operation execute in a single cycle and are useful for scaling data in parallel with other operations.

Three main addressing modes are available with the instruction set: direct, indirect, and immediate addressing.

#### direct addressing

In direct addressing, seven bits of the instruction word concatenated with the 1-bit data page pointer form the data memory address. This implements a paging scheme in which the first page contains 128 words, and the second page contains up to 128 words.

#### indirect addressing

Indirect addressing forms the data memory address from the least-significant eight bits of one of the two auxiliary registers, ARO and AR1. The Auxiliary Register Pointer (ARP) selects the current auxiliary register. The auxiliary registers can be automatically incremented or decremented and the ARP changed in parallel with the execution of any indirect instruction to permit single-cycle manipulation of data tables. Indirect addressing can be used with all instructions requiring data operands, except for the immediate operand instructions.

#### immediate addressing

Immediate instructions derive data from part of the instruction word rather than from the data RAM. Some useful immediate instructions are multiply immediate (MPYK), load accumulator immediate (LACK), and load auxiliary register immediate (LARK).

#### instruction set summary

Table 2 lists the symbols and abbreviations used in Table 3, the instruction set summary. Table 3 contains a short description and the opcode for each TMS320 first-generation instruction. The summary is arranged according to function and alphabetized within each functional group.

TABLE 2. INSTRUCTION SYMBOLS

SYMBOL	MEANING
ACC	Accumulator
D	Data memory address field
I	Addressing mode bit
K	Immediate operand field
PA	3-bit port address field
R	1-bit operand field specifying auxiliary register
S	4-bit left-shift code
X	3-bit accumulator left-shift field

TEXAS  
INSTRUMENTS

P.O. BOX 604 1443 • HOUSTON, TEXAS 77001

TMS320 FIRST-GENERATION  
DEVICES

TABLE 3. TMS320 FIRST-GENERATION INSTRUCTION SET SUMMARY

ACCUMULATOR INSTRUCTIONS																		
MNEMONIC	DESCRIPTION	NO. CYCLES	NO WORDS	OPCODE INSTRUCTION REGISTER														
				15	14	13	12	11	10	9	8	7	6	5	4	3	2	1
ABS	Absolute value of accumulator	1	1	0	1	1	1	1	1	1	1	1	0	0	0	1	0	0
ADD	Add to accumulator with shift	1	1	0	0	0	0	←S	→	1	←D	→	←D	→	←D	→	←D	→
ADDH	Add to high-order accumulator bits	1	1	0	1	1	0	0	0	0	1	←D	→	←D	→	←D	→	←D
ADDS	Add to accumulator with no sign extension	1	1	0	1	1	0	0	0	0	1	1	←D	→	←D	→	←D	→
AND	AND with accumulator	1	1	0	1	1	1	1	0	0	1	1	←D	→	←D	→	←D	→
LAC	Load accumulator with shift	1	1	0	0	1	0	←S	→	1	←D	→	←D	→	←D	→	←D	→
LACK	Load accumulator immediate	1	1	0	1	1	1	1	1	1	0	←K	→	←D	→	←D	→	←D
OR	OR with accumulator	1	1	0	1	1	1	1	0	1	0	1	←D	→	←D	→	←D	→
SACH	Store high-order accumulator bits with shift	1	1	0	1	0	1	1	←X	→	1	←D	→	←D	→	←D	→	←D
SACL	Store low-order accumulator bits	1	1	0	1	0	1	0	0	0	0	1	←D	→	←D	→	←D	→
SUB	Subtract from accumulator with shift	1	1	0	0	0	1	←S	→	1	←D	→	←D	→	←D	→	←D	→
SUBC	Conditional subtract (for divide)	1	1	0	1	1	0	0	1	0	0	1	←D	→	←D	→	←D	→
SUBH	Subtract from high-order accumulator bits	1	1	0	1	1	0	0	0	1	0	1	←D	→	←D	→	←D	→
SUBS	Subtract from accumulator with no sign extension	1	1	0	1	1	0	0	0	1	1	1	←D	→	←D	→	←D	→
XOR	Exclusive OR with accumulator	1	1	0	1	1	1	1	0	0	0	1	←D	→	←D	→	←D	→
ZAC	Zero accumulator	1	1	0	1	1	1	1	1	1	1	1	0	0	0	1	0	0
ZALH	Zero accumulator and load high-order bits	1	1	0	1	1	0	0	1	0	1	1	←D	→	←D	→	←D	→
ZALS	Zero accumulator and load low-order bits with no sign extension	1	1	0	1	1	0	0	1	1	0	1	←D	→	←D	→	←D	→
AUXILIARY REGISTER AND DATA PAGE POINTER INSTRUCTIONS																		
MNEMONIC	DESCRIPTION	NO. CYCLES	NO. WORDS	OPCODE INSTRUCTION REGISTER														
				15	14	13	12	11	10	9	8	7	6	5	4	3	2	1
LAR	Load auxiliary register	1	1	0	0	1	1	1	0	0	0	1	←D	→	←D	→	←D	→
LARK	Load auxiliary register immediate	1	1	0	1	1	1	0	0	0	0	←K	→	←D	→	←D	→	←D
LARP	Load auxiliary register pointer immediate	1	1	0	1	1	0	1	0	0	0	1	0	0	0	0	0	K
LDP	Load data memory page pointer	1	1	0	1	1	0	1	1	1	1	1	←D	→	←D	→	←D	→
LDPK	Load data memory page pointer immediate	1	1	0	1	1	0	1	1	1	0	0	0	0	0	0	0	K
MAR	Modify auxiliary register and pointer	1	1	1	1	0	1	0	0	0	0	1	←D	→	←D	→	←D	→
SAR	Store auxiliary register	1	1	0	0	1	1	0	0	0	0	1	←D	→	←D	→	←D	→



TEXAS  
INSTRUMENTS

POST OFFICE BOX 1443 • HOUSTON, TEXAS 77001

# TMS320 FIRST-GENERATION DEVICES

TABLE 3. TMS320 FIRST-GENERATION INSTRUCTION SET SUMMARY (CONTINUED)

BRANCH INSTRUCTIONS				
MNEMONIC	DESCRIPTION	NO. CYCLES	NO. WORDS	OPCODE
				INSTRUCTION REGISTER
				15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
B	Branch unconditional	2	2	1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0
BANZ	Branch on auxiliary register not zero	2	2	0 0 0 0 ← BRANCH ADDRESS →
BGEZ	Branch if accumulator $\geq 0$	2	2	1 1 1 1 0 1 0 0 0 0 0 0 0 0 0 0
BGZ	Branch if accumulator $> 0$	2	2	0 0 0 0 ← BRANCH ADDRESS →
BIOZ	Branch if $\overline{EO} = 0$	2	2	1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0
BLEZ	Branch if accumulator $\leq 0$	2	2	0 0 0 0 ← BRANCH ADDRESS →
BLZ	Branch if accumulator $< 0$	2	2	1 1 1 1 1 0 1 0 0 0 0 0 0 0 0 0
BNZ	Branch if accumulator $\neq 0$	2	2	0 0 0 0 ← BRANCH ADDRESS →
BOV	Branch on overflow	2	2	1 1 1 1 0 1 0 0 0 0 0 0 0 0 0 0
BZ	Branch if accumulator $= 0$	2	2	0 0 0 0 ← BRANCH ADDRESS →
CALA	Call subroutine from accumulator	2	1	0 1 1 1 1 1 1 1 1 0 0 0 1 1 0 0
CALL	Call subroutine immediately	2	2	1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0
RET	Return from subroutine or interrupt routine	2	1	0 0 0 0 ← BRANCH ADDRESS →
				0 1 1 1 1 1 1 1 1 0 0 0 1 1 0 1
T REGISTER, P REGISTER, AND MULTIPLY INSTRUCTIONS				
MNEMONIC	DESCRIPTION	NO. CYCLES	NO. WORDS	OPCODE
				INSTRUCTION REGISTER
				15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
APAC	Add P register to accumulator	1	1	0 1 1 1 1 1 1 1 1 0 0 0 1 1 1 1
LT	Load T register	1	1	0 1 1 0 1 0 1 0 1 ← D →
LTA	LTA combines LT and APAC into one instruction	1	1	0 1 1 0 1 1 0 0 1 ← D →
LTD	LTD combines LT, APAC, and DMV into one instruction	1	1	0 1 1 0 1 0 1 1 ← D →
MPV	Multiply with T register; store product in P register	1	1	0 1 1 0 1 1 0 1 ← D →
MPVK	Multiply T register with immediate operand; store product in P register	1	1	1 0 0 ← K →
PAC	Load accumulator from P register	1	1	0 1 1 1 1 1 1 1 1 0 0 0 1 1 1 0
SPAC	Subtract P register from accumulator	1	1	0 1 1 1 1 1 1 1 1 0 0 1 0 0 0 0

TEXAS  
INSTRUMENTS

POST OFFICE BOX 1443 • HOUSTON, TEXAS 77001

TMS320 FIRST-GENERATION  
DEVICES

TABLE 3. TMS320 FIRST-GENERATION INSTRUCTION SET SUMMARY (CONCLUDED)

CONTROL INSTRUCTIONS																			
MNEMONIC	DESCRIPTION	NO. CYCLES	NO WORDS	OPCODE															
				INSTRUCTION REGISTER															
				15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DINT	Disable interrupt	1	1	0	1	1	1	1	1	1	1	1	0	0	0	0	0	0	1
EINT	Enable interrupt	1	1	0	1	1	1	1	1	1	1	1	0	0	0	0	0	1	0
LST	Load status register	1	1	0	1	1	1	1	0	1	1	1	← D →						
NOP	No operation	1	1	0	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0
POP	POP stack to accumulator	2	1	0	1	1	1	1	1	1	1	1	0	0	1	1	1	0	1
PUSH	PUSH stack from accumulator	2	1	0	1	1	1	1	1	1	1	1	0	0	1	1	1	0	0
ROVM	Reset overflow mode	1	1	0	1	1	1	1	1	1	1	1	0	0	0	1	0	1	0
SOVM	Set overflow mode	1	1	0	1	1	1	1	1	1	1	1	0	0	0	1	0	1	1
SST	Store status register	1	1	0	1	1	1	1	1	0	0	1	← D →						
I/O AND DATA MEMORY OPERATIONS																			
MNEMONIC	DESCRIPTION	NO. CYCLES	NO. WORDS	OPCODE															
				INSTRUCTION REGISTER															
				15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DMOV	Copy contents of data memory location into next higher location	1	1	0	1	1	0	1	0	0	1	1	← D →						
IN	Input data from port	2	1	0	1	0	0	0	← PA →		1	← D →							
OUT	Output data to port	2	1	0	1	0	0	1	← PA →		1	← D →							
TBLR	Table read from program memory to data RAM	3	1	0	1	1	0	0	1	1	1	1	← D →						
TBLW	Table write from data RAM to program memory	3	1	0	1	1	1	1	1	0	1	1	← D →						

## development support

Texas Instruments offers an extensive line of development support products to assist the user in all aspects of TMS320 first-generation-based design and development. These products range from development and application software to complete hardware development and evaluation systems such as the XDS/22. Table 4 lists the development support products for the first-generation TMS320 devices.

System development begins with the use of the Evaluation Module (EVM) or Emulator (XDS). These hardware tools allow the designer to evaluate the processor's performance, benchmark time-critical code, and determine the feasibility of using a TMS320 device to implement a specific algorithm.

Software and hardware can be developed in parallel by using the macro assembler/linker and simulator for software development and the XDS for hardware development. The assembler/linker translates the system's assembly source program into an object module that can be executed by the simulator, XDS, or EVM. The XDS provides realtime in-circuit emulation and is a powerful tool for debugging and integrating software and hardware modules.

Additional support for the TMS320 products consists of extensive documentation and three-day DSP design workshops offered by the TI Regional Technology Centers (RTCs). The workshops provide hands-on experience with the TMS320 development tools. Refer to the *TMS320 Family Development Support Reference Guide* for further information about TMS320 development support products and DSP workshops. When technical questions arise regarding the TMS320, contact the Texas Instruments TMS320 DSP Hotline, (713) 274-2320.

TEXAS  
INSTRUMENTS

POST OFFICE BOX 1443 • HOUSTON, TEXAS 77001



# TMS320 FIRST-GENERATION NMOS DEVICES

## NMOS DEVICE ELECTRICAL SPECIFICATIONS

This section contains all the electrical specifications for the TMS320 NMOS first-generation devices. Refer to the top corner for the specific device.

absolute maximum ratings over specified temperature range (unless otherwise noted)<sup>†</sup>

Supply voltage range, $V_{CC}^{\ddagger}$	-0.3 V to 7 V
Input voltage range	-0.3 V to 15 V
Output voltage range	-0.3 V to 15 V
Continuous power dissipation	1.5 W
Air temperature range above operating device	0°C to 70°C
Storage temperature range	-55°C to +150°C

<sup>†</sup>Stresses beyond those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only, and functional operation of the device or any other conditions beyond those indicated in the "Recommended Operating Conditions" section of this specification is not implied. Exposure to absolute-maximum-rated conditions for extended periods may affect device reliability.

<sup>‡</sup>All voltage values are with respect to  $V_{SS}$ .

recommended operating conditions

		MIN	NOM	MAX	UNIT
VCC	Supply voltage	4.75	5	5.25	V
VSS	Supply voltage		0		V
VIH	High-level input voltage	All inputs except CLKIN			V
		CLKIN			
VIL	Low-level input voltage (all inputs)			0.8	V
IOH	High-level output current (all outputs)			-300	μA
IOL	Low-level output current (all outputs)			2	mA
TA	Operating free-air temperature	0		70	°C

electrical characteristics over specified temperature range (unless otherwise noted)

PARAMETER	TEST CONDITIONS	MIN	TYP <sup>†</sup>	MAX	UNIT
$V_{OH}$ High-level output voltage	$I_{OH} = \text{MAX}$	2.4	3		V
$V_{OL}$ Low-level output voltage	$I_{OL} = \text{MAX}$		0.3	0.5	V
$I_{OZ}$ Off-state output current	$V_{CC} = \text{MAX}$			20	$\mu$ A
				-20	$\mu$ A
$I_I$ Input current	$V_I = V_{SS} \text{ to } V_{CC}$			$\pm 20$	$\mu$ A
				$\pm 50$	$\mu$ A
$I_{CC}^{\ddagger}$ Supply current	$V_{CC} = \text{MAX}$		180	275	mA
				235 <sup>§</sup>	mA
$C_i$ Input capacitance	Data bus		25 <sup>§</sup>		pF
	All others		15 <sup>§</sup>		pF
$C_o$ Output capacitance	Data bus		25 <sup>§</sup>		pF
	All others		10 <sup>§</sup>		pF

<sup>†</sup>All typical values except for  $I_{CC}$  are at  $V_{CC} = 5$  V,  $T_A = 25^\circ\text{C}$ .

<sup>‡</sup> $I_{CC}$  characteristics are inversely proportional to temperature; i.e.,  $I_{CC}$  decreases approximately linearly with temperature.

<sup>§</sup>Value derived from characterization data and not tested.

# TMS320 FIRST-GENERATION NMOS DEVICES

## PARAMETER MEASUREMENT INFORMATION

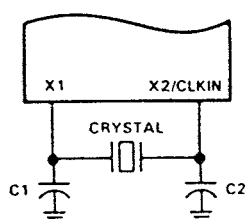


FIGURE 1. INTERNAL CLOCK OPTION

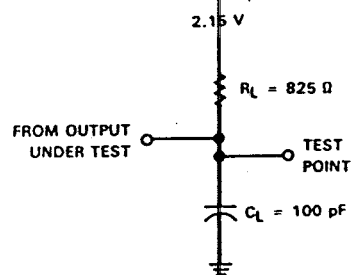


FIGURE 2. TEST LOAD CIRCUIT

### input synchronization requirements

For systems using asynchronous inputs to the  $\overline{\text{INT}}$  and  $\overline{\text{BIO}}$  pins on the TMS32010, the external hardware shown in the diagrams below is recommended to ensure proper execution of interrupts and the BIOZ instruction. This hardware synchronizes the  $\overline{\text{INT}}$  and  $\overline{\text{BIO}}$  input signals with the rising edge of CLKOUT on the TMS32010. The pulse width required for these input signals is  $t_{\text{C}}(\text{C})$ , which is one TMS32010 clock cycle, plus sufficient setup time for the flip-flop (dependent upon the flip-flop used). Note that these input synchronization requirements apply only to NMOS versions of the TMS32010 and not to other members of the TMS320 family.

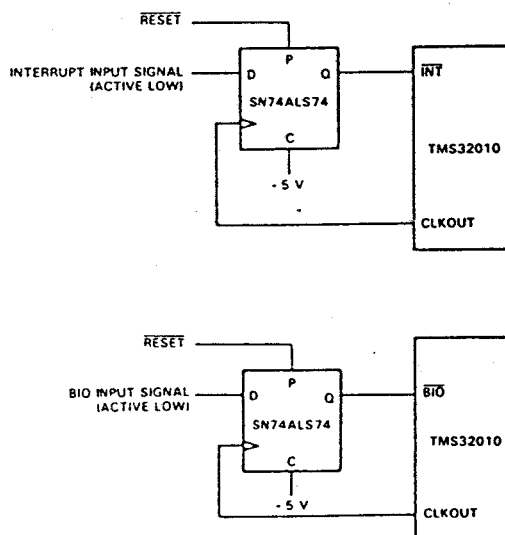


FIGURE 3. ASYNCHRONOUS INPUT SYNCHRONIZATION CIRCUITS

TEXAS  
INSTRUMENTS

POST OFFICE BOX 1443 • HOUSTON, TEXAS 77001

**TMS32010**  
**TMS32010-14**
**CLOCK CHARACTERISTICS AND TIMING**

The TMS32010 can use either its internal oscillator or an external frequency source for a clock.

**internal clock option**

The internal oscillator is enabled by connecting a crystal across X1 and X2/CLKIN (see Figure 1). The frequency of CLKOUT is one-fourth the crystal fundamental frequency. The crystal should be fundamental mode, and parallel resonant, with an effective series resistance of 30 ohms, a power dissipation of 1 mW, and be specified at a load capacitance of 20 pF.

PARAMETER	TEST CONDITIONS	TMS32010			TMS32010-14			UNIT
		MIN	NOM	MAX	MIN	NOM	MAX	
Crystal frequency $f_x$	0°C to 70°C	6.7		20.5	6.7		14.4	MHz
C1, C2	0°C to 70°C		10			10		pF

**external clock option**

An external frequency source can be used by injecting the frequency directly into X2/CLKIN with X1 left unconnected. The external frequency injected must conform to the specifications listed in the table below.

**switching characteristics over recommended operating conditions**

PARAMETER	TEST CONDITIONS	TMS32010			TMS32010-14			UNIT
		MIN	NOM	MAX	MIN	NOM	MAX	
$t_{c(C)}$ CLKOUT cycle time <sup>1</sup>	$R_L = 825 \Omega$ , $C_L = 100 \text{ pF}$ , See Figure 2	195.12	200		277.78			ns
$t_{r(C)}$ CLKOUT rise time			10			10		ns
$t_{f(C)}$ CLKOUT fall time			8			8		ns
$t_{w(CL)}$ Pulse duration, CLKOUT low			92			131		ns
$t_{w(CH)}$ Pulse duration, CLKOUT high			90			129		ns
$t_{d(MCC)}$ Delay time CLKIN to CLKOUT <sup>1</sup>		25 <sup>2</sup>		60 <sup>2</sup>	25 <sup>2</sup>		60 <sup>2</sup>	ns

<sup>1</sup>  $t_{c(C)}$  is the cycle time of CLKOUT, i.e.,  $4 \cdot t_{c(MC)}$  (4 times CLKIN cycle time if an external oscillator is used).

<sup>2</sup> Values derived from characterization data and not tested.

**timing requirements over recommended operating conditions**

		TMS32010			TMS32010-14			UNIT
		MIN	NOM	MAX	MIN	NOM	MAX	
$t_{c(MC)}$ Master clock cycle time		46.78	50	150	69.5		150	ns
$t_{r(MC)}$ Rise time master clock input			5 <sup>†</sup>	10 <sup>†</sup>		5 <sup>†</sup>	10 <sup>†</sup>	ns
$t_{f(MC)}$ Fall time master clock input			5 <sup>†</sup>	10 <sup>†</sup>		5 <sup>†</sup>	10 <sup>†</sup>	ns
$t_{w(MCP)}$ Pulse duration master clock		$0.475t_{c(MC)}$ <sup>†</sup>		$0.525t_{c(MC)}$ <sup>†</sup>	$0.475t_{c(MC)}$ <sup>†</sup>		$0.525t_{c(MC)}$ <sup>†</sup>	ns
$t_{w(MCL)}$ Pulse duration master clock low, $t_{c(MC)} = 50 \text{ ns}$			20 <sup>†</sup>			20 <sup>†</sup>		ns
$t_{w(MCH)}$ Pulse duration master clock high, $t_{c(MC)} = 50 \text{ ns}$			20 <sup>†</sup>			20 <sup>†</sup>		ns

<sup>†</sup> Values derived from characterization data and not tested.

TEXAS  
INSTRUMENTS

POST OFFICE BOX 1443 • HOUSTON, TEXAS 77001

TMS32010  
TMS32010-14

### RESET ( $\overline{RS}$ ) TIMING

Switching characteristics over recommended operating conditions

PARAMETER	TEST CONDITIONS	MIN	TYP	MAX	UNIT
$t_{d11}$ Delay time $\overline{DEN}1$ , $\overline{WE}1$ , and $\overline{MEN}1$ from $\overline{RS}$	$R_L = 825 \Omega$ $C_L = 100 \text{ pF}$			$\frac{1}{2}t_{c(C)} + 50^1$	ns
$t_{dis(R)}$ Data bus disable time after $\overline{RS}$	See Figure 2			$\frac{1}{2}t_{c(C)} + 50^1$	ns

<sup>1</sup>Values derived from characterization data and not tested.

Timing requirements over recommended operating conditions

	TMS32010			TMS32010-14			UNIT
	MIN	NOM	MAX	MIN	NOM	MAX	
$t_{su(R)}$ Reset $\overline{RS}$ setup time prior to CLKOUT (see Note 3)	50			50			ns
$t_w(R)$ $\overline{RS}$ pulse duration	$5t_{c(C)}$			$5t_{c(C)}$			ns

NOTE 3:  $\overline{RS}$  can occur anytime during a clock cycle. Time given is minimum to ensure synchronous operation.

### INTERRUPT ( $\overline{INT}$ ) TIMING

Timing requirements over recommended operating conditions

	TMS32010			TMS32010-14			UNIT
	MIN	NOM	MAX	MIN	NOM	MAX	
$t_f(\overline{INT})$ Fall time ( $\overline{INT}$ )			15			15	ns
$t_w(\overline{INT})$ Pulse duration $\overline{INT}$	$t_{c(C)}$			$t_{c(C)}$			ns
$t_{su}(\overline{INT})$ Setup time $\overline{INT}1$ before CLKOUT.	50			50			ns

### I/O ( $\overline{BIO}$ ) TIMING

Timing requirements over recommended operating conditions

	TMS32010			TMS32010-14			UNIT
	MIN	NOM	MAX	MIN	NOM	MAX	
$t_f(\overline{BIO})$ Fall time $\overline{BIO}$			15			15	ns
$t_w(\overline{BIO})$ Pulse duration $\overline{BIO}$	$t_{c(C)}$			$t_{c(C)}$			ns
$t_{su}(\overline{BIO})$ Setup time $\overline{BIO}1$ before CLKOUT.	50			50			ns

TEXAS  
INSTRUMENTS

POST OFFICE BOX 1443 • HOUSTON, TEXAS 77001

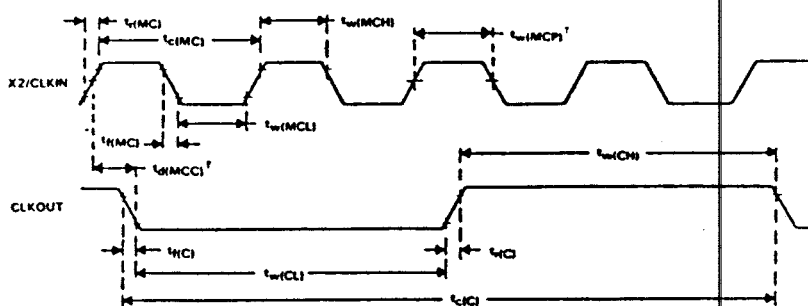
# TMS320 FIRST-GENERATION DEVICES

## TIMING DIAGRAMS

This section contains all the timing diagrams for the TMS320 first-generation devices. Refer to the top corner for the specific device.

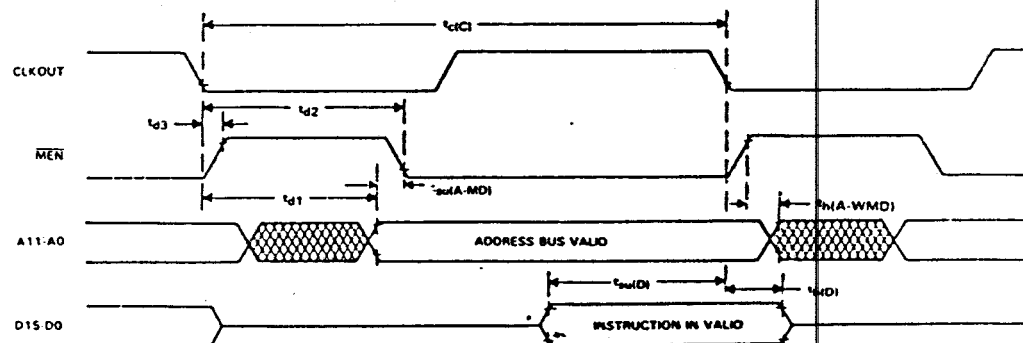
Timing measurements are referenced to and from a low voltage of 0.8 volts and a high voltage of 2.0 volts, unless otherwise noted.

### clock timing



$t_{1d}(MCC)$  and  $t_{2d}(MCP)$  are referenced to an intermediate level of 1.5 volts on the CLKIN waveform.

### memory read timing

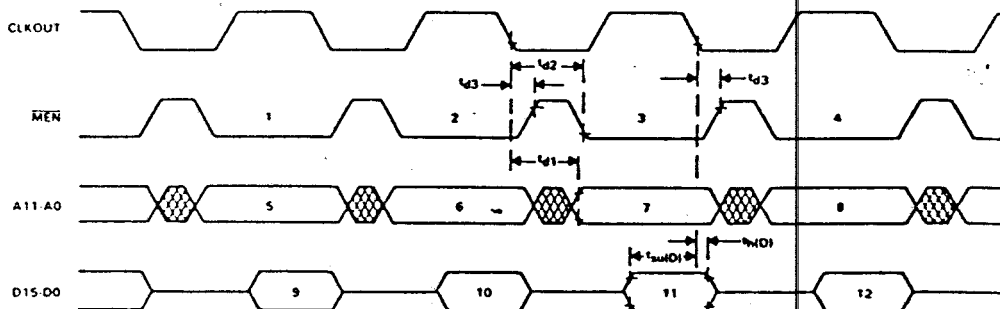


TEXAS  
INSTRUMENTS

POST OFFICE BOX 1443 • HOUSTON, TEXAS 77001

# TMS320 FIRST-GENERATION DEVICES

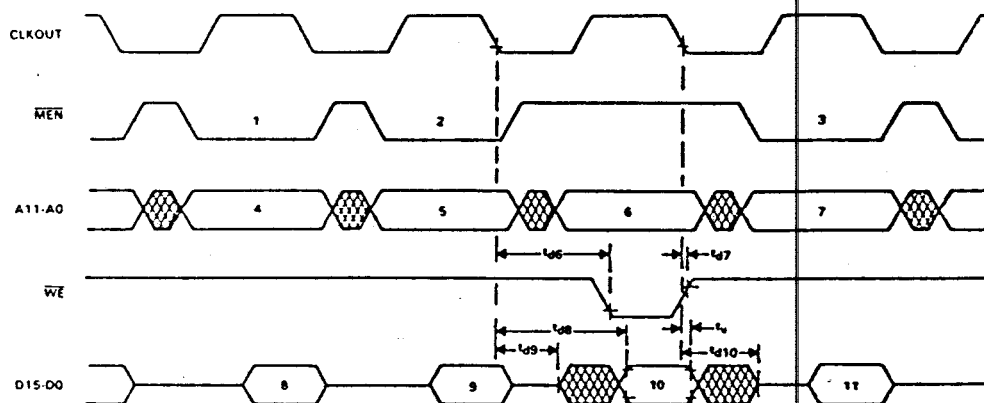
## TBLR instruction timing



### LEGEND:

- |                              |                          |
|------------------------------|--------------------------|
| 1. TBLR INSTRUCTION PREFETCH | 7. ADDRESS BUS VALID     |
| 2. DUMMY PREFETCH            | 8. ADDRESS BUS VALID     |
| 3. DATA FETCH                | 9. INSTRUCTION IN VALID  |
| 4. NEXT INSTRUCTION PREFETCH | 10. INSTRUCTION IN VALID |
| 5. ADDRESS BUS VALID         | 11. DATA IN VALID        |
| 6. ADDRESS BUS VALID         | 12. INSTRUCTION IN VALID |

## TBLW instruction timing

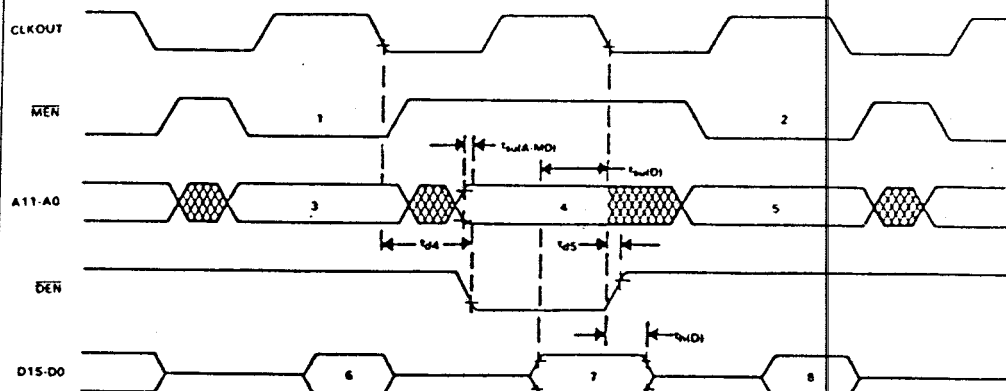


### LEGEND:

- |                              |                          |
|------------------------------|--------------------------|
| 1. TBLW INSTRUCTION PREFETCH | 7. ADDRESS BUS VALID     |
| 2. DUMMY PREFETCH            | 8. INSTRUCTION IN VALID  |
| 3. NEXT INSTRUCTION PREFETCH | 9. INSTRUCTION IN VALID  |
| 4. ADDRESS BUS VALID         | 10. DATA OUT VALID       |
| 5. ADDRESS BUS VALID         | 11. INSTRUCTION IN VALID |
| 6. ADDRESS BUS VALID         |                          |

# TMS320 FIRST-GENERATION DEVICES

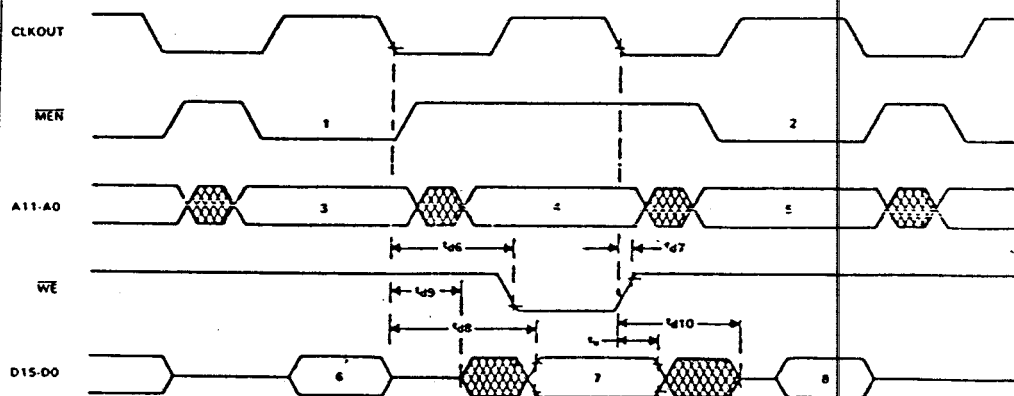
## IN instruction timing



### LEGEND:

- |                              |                         |
|------------------------------|-------------------------|
| 1. IN INSTRUCTION PREFETCH   | 5. ADDRESS BUS VALID    |
| 2. NEXT INSTRUCTION PREFETCH | 6. INSTRUCTION IN VALID |
| 3. ADDRESS BUS VALID         | 7. DATA IN VALID        |
| 4. PERIPHERAL ADDRESS VALID  | 8. INSTRUCTION IN VALID |

## OUT instruction timing



### LEGEND:

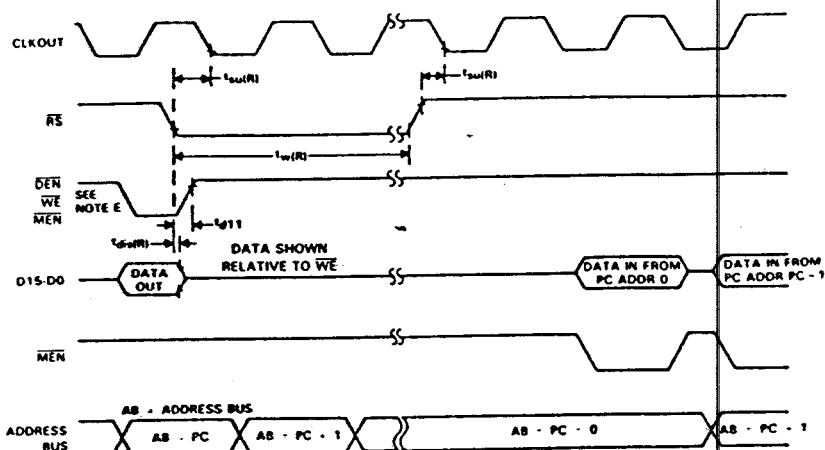
- |                              |                         |
|------------------------------|-------------------------|
| 1. OUT INSTRUCTION PREFETCH  | 5. ADDRESS BUS VALID    |
| 2. NEXT INSTRUCTION PREFETCH | 6. INSTRUCTION IN VALID |
| 3. ADDRESS BUS VALID         | 7. DATA IN VALID        |
| 4. PERIPHERAL ADDRESS VALID  | 8. INSTRUCTION IN VALID |

TEXAS  
INSTRUMENTS

POST OFFICE BOX 1443 • HOUSTON, TEXAS 77001

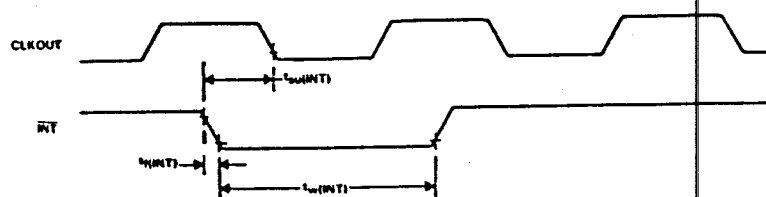
**TMS32010, TMS32010-14**  
**TMS320C10, TMS320C10-25, TMS320C10-14**  
**TMS320C15/E15, TMS320C15-25**

**reset timing**

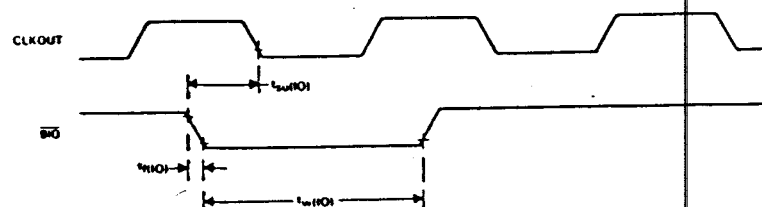


- NOTES: A.  $\overline{RS}$  forces  $\overline{DEN}$ ,  $\overline{WE}$ , and  $\overline{MEN}$  high and places data bus D0 through D15 in a high-impedance state. AB outputs (and program counter) are synchronously cleared to zero after the next complete CLK cycle from  $\overline{RS}$ .
- B.  $\overline{RS}$  must be maintained for a minimum of five clock cycles.
- C. Resumption of normal program will commence after one complete CLK cycle from  $\overline{RS}$ .
- D. Due to the synchronizing action on  $\overline{RS}$ , time to execute the function can vary dependent upon when  $\overline{RS}$  or  $\overline{RS}$  occur in the CLK cycle.
- E. Diagram shown is for definition purpose only.  $\overline{DEN}$ ,  $\overline{WE}$ , and  $\overline{MEN}$  are mutually exclusive.
- F. During a write cycle,  $\overline{RS}$  may produce an invalid write address.

**interrupt timing**



**BIO timing**



TEXAS  
INSTRUMENTS

POST OFFICE BOX 1443 • HOUSTON, TEXAS 77001





## ADC0801, ADC0802, ADC0803, ADC0804, ADC0805 8-Bit $\mu$ P Compatible A/D Converters

### General Description

The ADC0801, ADC0802, ADC0803, ADC0804 and ADC0805 are CMOS 8-bit successive approximation A/D converters which use a differential potentiometric ladder—similar to the 256R products. These converters are designed to allow operation with the NSC800 and INS8080A derivative control bus, and TRI-STATE<sup>®</sup> output latches directly drive the data bus. These A/Ds appear like memory locations or I/O ports to the microprocessor and no interfacing logic needed.

A new differential analog voltage input allows increasing the common-mode rejection and offsetting the analog zero input voltage value. In addition, the voltage reference input can be adjusted to allow encoding any smaller analog voltage span to the full 8 bits of resolution.

### Features

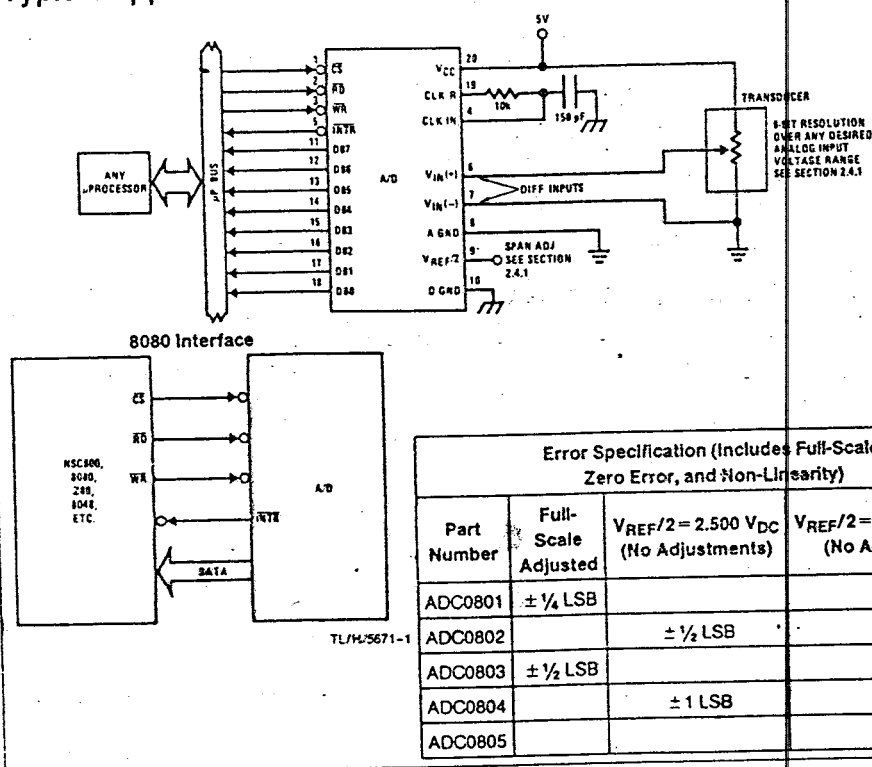
- Compatible with 8080  $\mu$ P derivatives—no interfacing logic needed - access time - 135 ns
- Easy interface to all microprocessors, or operates "stand alone"

- Differential analog voltage inputs
- Logic inputs and outputs meet both MOS and T<sup>2</sup>L voltage level specifications
- Works with 2.5V (LM336) voltage reference
- On-chip clock generator
- 0V to 5V analog input voltage range with single 5V supply
- No zero adjust required
- 0.3" standard width 20-pin DIP package
- Operates ratiometrically or with 5 V<sub>DC</sub>, 2.5 V<sub>DC</sub>, or analog span adjusted voltage reference

### Key Specifications

- Resolution 8 bits
- Total error  $\pm 1/4$  LSB,  $\pm 1/2$  LSB and  $\pm 1$  LSB
- Conversion time 100  $\mu$ s

### Typical Applications



**Absolute Maximum Ratings** (Notes 1 & 2)

Supply Voltage ( $V_{CC}$ ) (Note 3)	6.5V
Voltage	
Logic Control Inputs	-1.5V to +18V
At Other Input and Outputs	-0.3V to ( $V_{CC} - 0.3$ )V
Storage Temperature Range	-65°C to +150°C
Package Dissipation at $T_A = 25^\circ\text{C}$	575 mW
Lead Temperature (Soldering, 10 seconds)	300°C

**Operating Ratings** (Notes 1 & 2)

Temperature Range	$T_{MIN} \leq T_A \leq T_{MAX}$ -55°C $\leq T_A \leq$ +125°C -40°C $\leq T_A \leq$ +85°C -40°C $\leq T_A \leq$ +85°C 0°C $\leq T_A \leq$ +70°C 4.5 V <sub>DC</sub> to 6.3 V <sub>DC</sub>
ADC0801/02LD	
ADC0801/02/03/04LCD	
ADC0801/02/03/05LCN	
ADC0804LCN	
Range of $V_{CC}$	

**Electrical Characteristics**

The following specifications apply for  $V_{CC} = 5\text{ V}_{DC}$ ,  $T_{MIN} \leq T_A \leq T_{MAX}$  and  $f_{CLK} = 640\text{ kHz}$  unless otherwise specified.

Parameter	Conditions	Min	Typ	Max	Units
ADC0801: Total Adjusted Error (Note 8)	With Full-Scale Adj. (See Section 2.5.2)			$\pm 1/4$	LSB
ADC0802: Total Unadjusted Error (Note 8)	$V_{REF}/2 = 2.500\text{ V}_{DC}$			$\pm 1/2$	LSB
ADC0803: Total Adjusted Error (Note 8)	With Full-Scale Adj. (See Section 2.5.2)			$\pm 1/2$	LSB
ADC0804: Total Unadjusted Error (Note 8)	$V_{REF}/2 = 2.500\text{ V}_{DC}$			$\pm 1$	LSB
ADC0805: Total Unadjusted Error (Note 8)	$V_{REF}/2$ -No Connection			$\pm 1$	LSB
$V_{REF}/2$ Input Resistance (Pin 9)	ADC0801/02/03/05 ADC0804 (Note 9)	2.5 1.0	8.0 1.3		k $\Omega$ k $\Omega$
Analog Input Voltage Range	(Note 4) $V(+)$ or $V(-)$	Gnd-0.05		$V_{CC} + 0.05$	V <sub>DC</sub>
DC Common-Mode Error	Over Analog Input Voltage Range		$\pm 1/16$	$\pm 1/8$	LSB
Power Supply Sensitivity	$V_{CC} = 5\text{ V}_{DC} \pm 10\%$ Over Allowed $V_{IN}(+)$ and $V_{IN}(-)$ Voltage Range (Note 4)		$\pm 1/16$	$\pm 1/8$	LSB

**AC Electrical Characteristics**

The following specifications apply for  $V_{CC} = 5\text{ V}_{DC}$  and  $T_A = 25^\circ\text{C}$  unless otherwise specified.

Parameter	Conditions	Min	Typ	Max	Units
$T_C$ Conversion Time	$f_{CLK} = 640\text{ kHz}$ (Note 6)	103		114	$\mu\text{s}$
$T_C$ Conversion Time	(Note 5, 6)	66		73	1/ $f_{CLK}$
$f_{CLK}$ Clock Frequency Clock Duty Cycle	$V_{CC} = 5\text{V}$ , (Note 5) (Note 5)	100 40	640	1460 60	kHz %
CR Conversion Rate in Free-Running Mode	$\overline{INTR}$ tied to $\overline{WR}$ with $\overline{CS} = 0\text{ V}_{DC}$ , $f_{CLK} = 640\text{ kHz}$			8770	conv/s
$t_W(\overline{WR})_L$ Width of $\overline{WR}$ Input (Start Pulse Width)	$\overline{CS} = 0\text{ V}_{DC}$ (Note 7)	100			ns
$t_{ACC}$ Access Time (Delay from Falling Edge of $\overline{RD}$ to Output Data Valid)	$C_L = 100\text{ pF}$		135	200	ns
$t_{IH}, t_{OH}$ TRI-STATE Control (Delay from Rising Edge of $\overline{RD}$ to Hi-Z State)	$C_L = 10\text{ pF}$ , $R_L = 10\text{k}$ (See TRI-STATE Test Circuits)		125	200	ns
$t_{WI}, t_{RI}$ Delay from Falling Edge of $\overline{WR}$ or $\overline{RD}$ to Reset of $\overline{INTR}$			300	450	ns
$C_{IN}$ Input Capacitance of Logic Control Inputs			5	7.5	pF
$C_{OUT}$ TRI-STATE Output Capacitance (Data Buffers)			5	7.5	pF

## Electrical Characteristics

The following specifications apply for  $V_{CC} = 5V_{DC}$  and  $T_{MIN} \leq T_A \leq T_{MAX}$ , unless otherwise specified.

Parameter	Conditions	Min	Typ	Max	Units
<b>CONTROL INPUTS</b> [Note: CLK IN (Pin 4) is the input of a Schmitt trigger circuit and is therefore specified separately]					
$V_{IN}(1)$	Logical "1" Input Voltage (Except Pin 4 CLK IN)	$V_{CC} = 5.25 V_{DC}$	2.0	15	$V_{DC}$
$V_{IN}(0)$	Logical "0" Input Voltage (Except Pin 4 CLK IN)	$V_{CC} = 4.75 V_{DC}$		0.8	$V_{DC}$
$I_{IN}(1)$	Logical "1" Input Current (All Inputs)	$V_{IN} = 5 V_{DC}$	0.005	1	$\mu A_{DC}$
$I_{IN}(0)$	Logical "0" Input Current (All Inputs)	$V_{IN} = 0 V_{DC}$	-1	-0.005	$\mu A_{DC}$
<b>CLOCK IN AND CLOCK R</b>					
$V_{T+}$	CLK IN (Pin 4) Positive Going Threshold Voltage		2.7	3.1	$V_{DC}$
$V_{T-}$	CLK IN (Pin 4) Negative Going Threshold Voltage		1.5	1.8	$V_{DC}$
$V_H$	CLK IN (Pin 4) Hysteresis ( $V_{T+} - V_{T-}$ )		0.6	1.3	$V_{DC}$
$V_{OUT}(0)$	Logical "0" CLK R Output Voltage	$I_O = 360 \mu A$ $V_{CC} = 4.75 V_{DC}$		0.4	$V_{DC}$
$V_{OUT}(1)$	Logical "1" CLK R Output Voltage	$I_O = -360 \mu A$ $V_{CC} = 4.75 V_{DC}$	2.4		$V_{DC}$
<b>DATA OUTPUTS AND INTR</b>					
$V_{OUT}(0)$	Logical "0" Output Voltage Data Outputs INTR Output	$I_{OUT} = 1.6 mA, V_{CC} = 4.75 V_{DC}$ $I_{OUT} = 1.0 mA, V_{CC} = 4.75 V_{DC}$		0.4 0.4	$V_{DC}$ $V_{DC}$
$V_{OUT}(1)$	Logical "1" Output Voltage	$I_O = -360 \mu A, V_{CC} = 4.75 V_{DC}$	2.4		$V_{DC}$
$V_{OUT}(1)$	Logical "1" Output Voltage	$I_O = -10 \mu A, V_{CC} = 4.75 V_{DC}$	4.5		$V_{DC}$
$I_{OUT}$	TRI-STATE Disabled Output Leakage (All Data Buffers)	$V_{OUT} = 0 V_{DC}$ $V_{OUT} = 5 V_{DC}$	-3		$\mu A_{DC}$ $\mu A_{DC}$
$I_{SOURCE}$		$V_{OUT}$ Short to Gnd, $T_A = 25^\circ C$	4.5	6	$mA_{DC}$
$I_{SINK}$		$V_{OUT}$ Short to $V_{CC}$ , $T_A = 25^\circ C$	9.0	16	$mA_{DC}$
<b>POWER SUPPLY</b>					
$I_{CC}$	Supply Current (Includes Ladder Current)	$f_{CLK} = 640 kHz$ , $V_{REF}/2 = NC$ , $T_A = 25^\circ C$ and $CS = "1"$ ADC0801/02/03/05 ADC0804 (Note 9)		1.1 1.9	1.8 2.5 $mA$ $mA$

Note 1: Absolute maximum ratings are those values beyond which the life of the device may be impaired.

Note 2: All voltages are measured with respect to Gnd, unless otherwise specified. The separate A Gnd point should always be wired to the D Gnd.

Note 3: A zener diode exists, internally, from  $V_{CC}$  to Gnd and has a typical breakdown voltage of  $7 V_{DC}$ .

Note 4: For  $V_{IN}(-) \geq V_{IN}(+)$  the digital output code will be 0000 0000. Two on-chip diodes are tied to each analog input (see block diagram) which will forward conduct for analog input voltages one diode drop below ground or one diode drop greater than the  $V_{CC}$  supply. Be careful, during testing at low  $V_{CC}$  levels (4.5V), as high level analog inputs (5V) can cause this input diode to conduct—especially at elevated temperatures, and cause errors for analog inputs near full-scale. The spec allows 50 mV forward bias of either diode. This means that as long as the analog  $V_{IN}$  does not exceed the supply voltage by more than 50 mV, the output code will be correct. To achieve an absolute 0  $V_{CC}$  to 5  $V_{DC}$  input voltage range will therefore require a minimum supply voltage of  $4.950 V_{DC}$  over temperature variations, initial tolerance and loading.

Note 5: Accuracy is guaranteed at  $f_{CLK} = 640 kHz$ . At higher clock frequencies accuracy can degrade. For lower clock frequencies, the duty cycle limits can be extended so long as the minimum clock high time interval or minimum clock low time interval is no less than 275 ns.

Note 6: With an asynchronous start pulse, up to 8 clock periods may be required before the internal clock phases are proper to start the conversion process. The start request is internally latched, see Figure 2 and section 2.0.

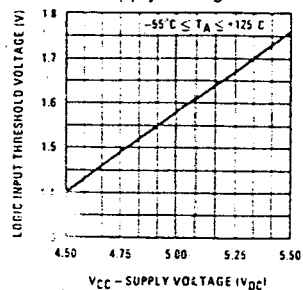
Note 7: The  $CS$  input is assumed to bracket the  $WR$  strobe input and therefore timing is dependent on the  $WR$  pulse width. An arbitrarily wide pulse width will hold the converter in a reset mode and the start of conversion is initiated by the low to high transition of the  $WR$  pulse (see timing diagrams).

Note 8: None of these A/Ds requires a zero adjust (see section 2.5.1). To obtain zero code at other analog input voltages see section 2.5 and Figure 5.

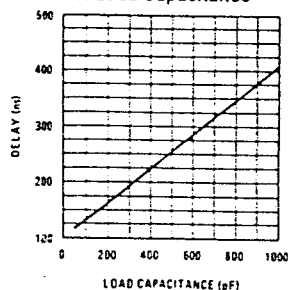
Note 9: For ADC0804LCD typical value of  $V_{REF}/2$  input resistance is 8 k $\Omega$  and of  $I_{CC}$  is 1.1 mA.

## Typical Performance Characteristics

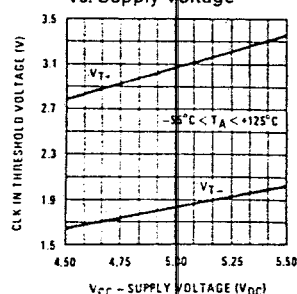
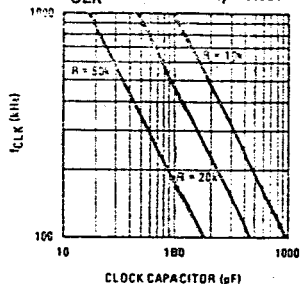
Logic Input Threshold Voltage vs. Supply Voltage



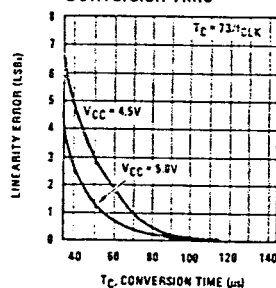
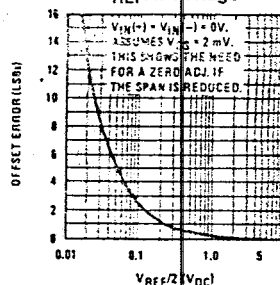
Delay From Falling Edge of RD to Output Data Valid vs. Load Capacitance



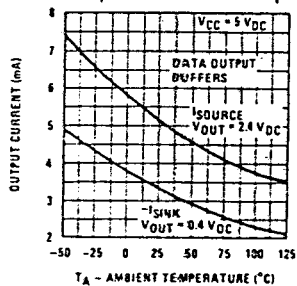
CLK IN Schmitt Trip Levels vs. Supply Voltage

f<sub>CLK</sub> vs. Clock Capacitor

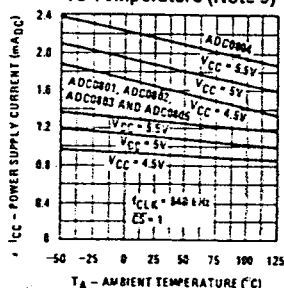
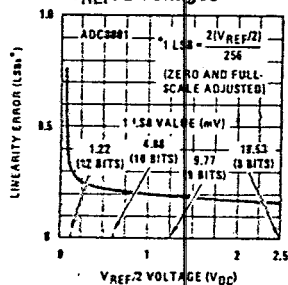
Full-Scale Error vs Conversion Time

Effect of Unadjusted Offset Error vs. V<sub>REF/2</sub> Voltage

Output Current vs Temperature

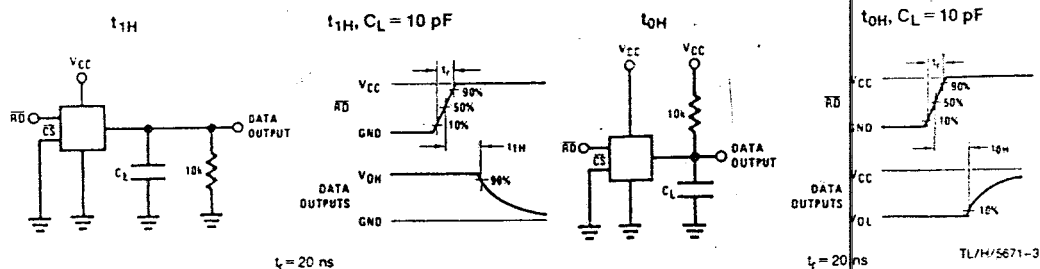


Power Supply Current vs Temperature (Note 9)

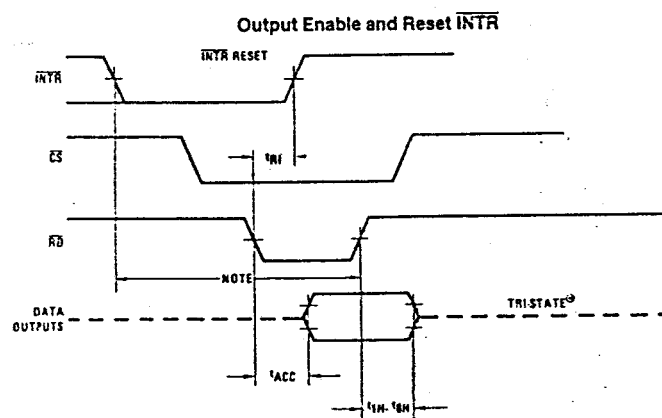
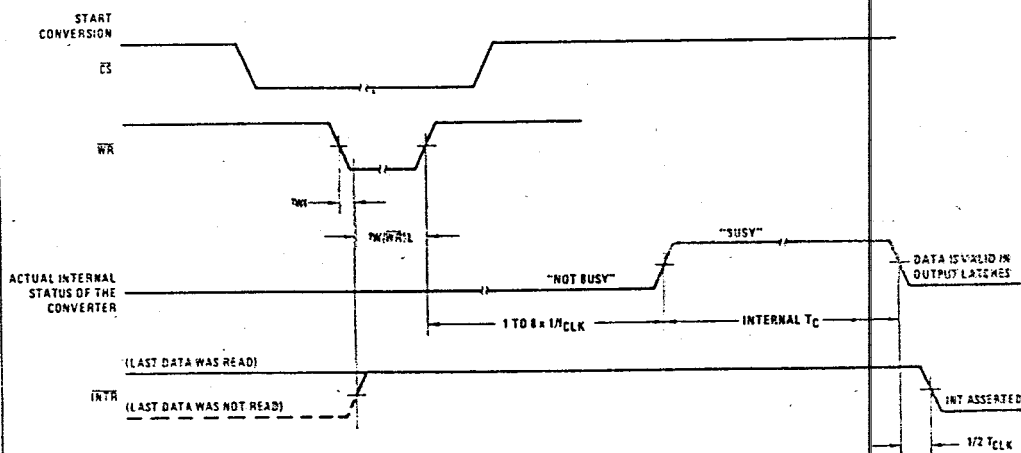
Linearity Error at Low V<sub>REF/2</sub> Voltages

TL/H/5671-2

# TRI-STATE Test Circuits and Waveforms



## Timing Diagrams (All timing is measured from the 50% voltage points)



Note: Read strobe must occur 8 clock periods ( $8/t_{CLK}$ ) after assertion of interrupt to guarantee reset of INTR.

TL/H/5671-4

### 1.0 UNDERSTANDING A/D ERROR SPECS

A perfect A/D transfer characteristic (staircase waveform) is shown in *Figure 1a*. The horizontal scale is analog input voltage and the particular points labeled are in steps of 1 LSB (19.53 mV with 2.5V tied to the  $V_{REF}/2$  pin). The digital output codes which correspond to these inputs are shown as  $D-1$ ,  $D$ , and  $D+1$ . For the perfect A/D, not only will center-value ( $A-1$ ,  $A$ ,  $A+1$ , ...) analog inputs produce the correct output digital codes, but also each riser (the transitions between adjacent output codes) will be located  $\pm 1/2$  LSB away from each center-value. As shown, the risers are ideal and have no width. Correct digital output codes will be provided for a range of analog input voltages which extend  $\pm 1/2$  LSB from the ideal center-values. Each tread (the range of analog input voltage which provides the same digital output code) is therefore 1 LSB wide.

*Figure 1b* shows a worst case error plot for the ADC0801. All center-valued inputs are guaranteed to produce the correct output codes and the adjacent risers are guaranteed to be no closer to the center-value points than  $\pm 1/4$  LSB. In other words, if we apply an analog input equal to the center-

value  $\pm 1/4$  LSB, we guarantee that the A/D will produce the correct digital code. The maximum range of the position of the code transition is indicated by the horizontal arrow and it is guaranteed to be no more than  $1/2$  LSB.

The error curve of *Figure 1c* shows a worst case error plot for the ADC0802. Here we guarantee that if we apply an analog input equal to the LSB analog voltage center-value the A/D will produce the correct digital code.

Next to each transfer function is shown the corresponding error plot. Many people may be more familiar with error plots than transfer functions. The analog input voltage to the A/D is provided by either a linear ramp or by the discrete output steps of a high resolution DAC. Notice that the error is continuously displayed and includes the quantization uncertainty of the A/D. For example the error at point 1 of *Figure 1a* is  $+1/2$  LSB because the digital code appeared  $1/2$  LSB in advance of the center-value of the tread. The error plots always have a constant negative slope and the abrupt up-side steps are always 1 LSB in magnitude.

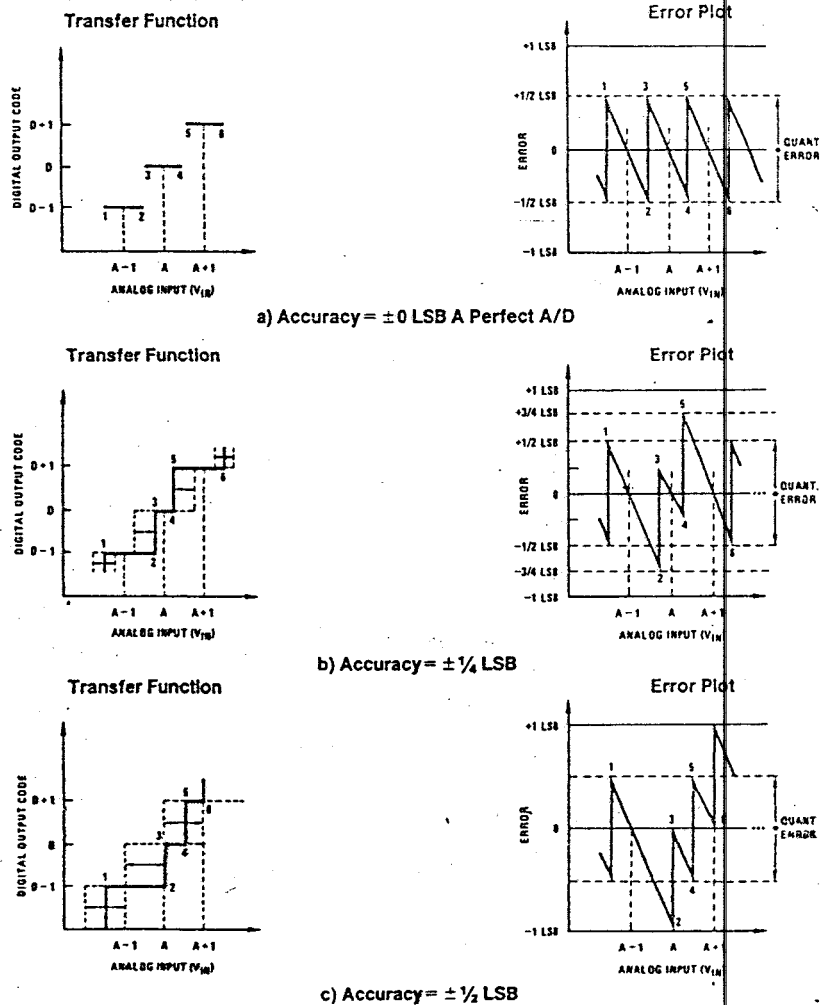


FIGURE 1. Clarifying the Error Specs of an A/D Converter

TL/H/5671-12

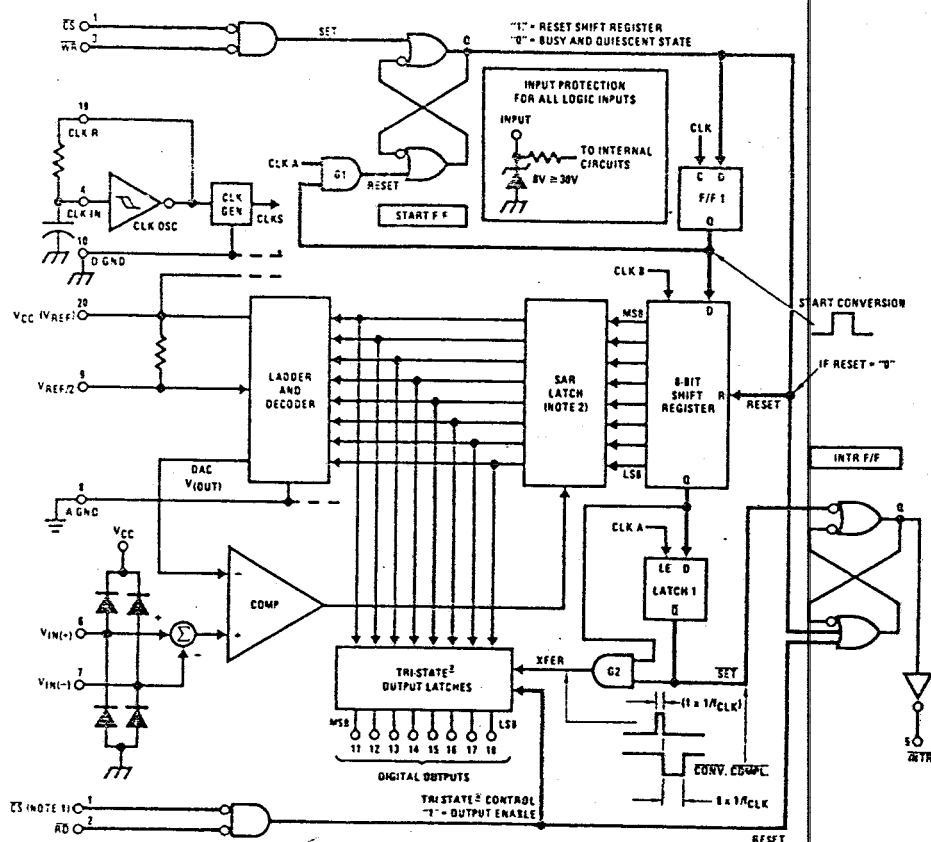
## 2.0 FUNCTIONAL DESCRIPTION

The ADC0801 series contains a circuit equivalent of the 256R network. Analog switches are sequenced by successive approximation logic to match the analog difference input voltage  $[V_{IN}(+) - V_{IN}(-)]$  to a corresponding tap on the R network. The most-significant bit is tested first and after 8 comparisons (64 clock cycles) a digital 8-bit binary code (1111 1111 = full-scale) is transferred to an output latch and then an interrupt is asserted (INTR makes a high-to-low transition). A conversion in process can be interrupted by issuing a second start command. The device may be operated in the free-running mode by connecting INTR to the  $\overline{WR}$  input with  $\overline{CS} = 0$ . To insure start-up under all possible conditions, an external  $\overline{WR}$  pulse is required during the first power-up cycle.

On the high-to-low transition of the  $\overline{WR}$  input the internal SAR latches and the shift register stages are reset. As long as the  $\overline{CS}$  input and  $\overline{WR}$  input remain low, the A/D will remain in a reset state. Conversion will start from 1 to 8 clock periods after at least one of these inputs makes a low-to-high transition.

A functional diagram of the A/D converter is shown in Figure 2. All of the package pinouts are shown and the major logic control paths are drawn in heavier weight lines.

The converter is started by having  $\overline{CS}$  and  $\overline{WR}$  simultaneously low. This sets the start flip-flop (F/F) and the resulting "1" level resets the 8-bit shift register, resets the Interrupt (INTR) F/F and inputs a "1" to the D flop, F/F1, which is at the input end of the 8-bit shift register. Internal clock signals then transfer this "1" to the Q output of F/F1. The AND gate, G1, combines this "1" output with a clock signal to provide a reset signal to the start F/F. If the set signal is no longer present (either  $\overline{WR}$  or  $\overline{CS}$  is a "1") the start F/F is reset and the 8-bit shift register then can have the "1" clocked in, which starts the conversion process. If the set signal were to still be present, this reset pulse would have no effect (both outputs of the start F/F would momentarily be at a "1" level) and the 8-bit shift register would continue to be held in the reset mode. This logic therefore allows for wide  $\overline{CS}$  and  $\overline{WR}$  signals and the converter will start after at least one of these signals returns high and the internal clocks again provide a reset signal for the start F/F.



Note 1:  $\overline{CS}$  shown twice for clarity.

Note 2: SAR = Successive Approximation Register.

FIGURE 2. Block Diagram

TL7H/5671-13

After the "1" is clocked through the 8-bit shift register (which completes the SAR search) it appears as the input to the D-type latch, LATCH 1. As soon as this "1" is output from the shift register, the AND gate, G2, causes the new digital word to transfer to the TRI-STATE output latches. When LATCH 1 is subsequently enabled, the Q output makes a high-to-low transition which causes the INTR F/F to set. An inverting buffer then supplies the  $\overline{\text{INTR}}$  input signal.

Note that this  $\overline{\text{SET}}$  control of the INTR F/F remains low for 8 of the external clock periods (as the internal clocks run at  $1/8$  of the frequency of the external clock). If the data output is continuously enabled ( $\overline{\text{CS}}$  and  $\overline{\text{RD}}$  both held low), the INTR output will still signal the end of conversion (by a high-to-low transition), because the  $\overline{\text{SET}}$  input can control the Q output of the INTR F/F even though the RESET input is constantly at a "1" level in this operating mode. This INTR output will therefore stay low for the duration of the  $\overline{\text{SET}}$  signal, which is 8 periods of the external clock frequency (assuming the A/D is not started during this interval).

When operating in the free-running or continuous conversion mode (INTR pin tied to  $\overline{\text{WR}}$  and  $\overline{\text{CS}}$  wired low—see also section 2.8), the START F/F is SET by the high-to-low transition of the INTR signal. This resets the SHIFT REGISTER which causes the input to the D-type latch, LATCH 1, to go low. As the latch enable input is still present, the  $\overline{\text{Q}}$  output will go high, which then allows the INTR F/F to be RESET. This reduces the width of the resulting INTR output pulse to only a few propagation delays (approximately 300 ns).

When data is to be read, the combination of both  $\overline{\text{CS}}$  and  $\overline{\text{RD}}$  being low will cause the INTR F/F to be reset and the TRI-STATE output latches will be enabled to provide the 8-bit digital outputs.

### 2.1 Digital Control Inputs

The digital control inputs ( $\overline{\text{CS}}$ ,  $\overline{\text{RD}}$ , and  $\overline{\text{WR}}$ ) meet standard T2L logic voltage levels. These signals have been renamed when compared to the standard A/D Start and Output Enable labels. In addition, these inputs are active low to allow an easy interface to microprocessor control busses. For non-microprocessor based applications, the  $\overline{\text{CS}}$  input (pin 1) can be grounded and the standard A/D Start function is obtained by an active low pulse applied at the  $\overline{\text{WR}}$  input (pin 3) and the Output Enable function is caused by an active low pulse at the  $\overline{\text{RD}}$  input (pin 2).

### 2.2 Analog Differential Voltage Inputs and Common-Mode Rejection

This A/D has additional applications flexibility due to the analog differential voltage input. The  $V_{\text{IN}}(-)$  input (pin 7) can be used to automatically subtract a fixed voltage value from the input reading (tare correction). This is also useful in 4 mA–20 mA current loop conversion. In addition, common-mode noise can be reduced by use of the differential input. The time interval between sampling  $V_{\text{IN}}(+)$  and  $V_{\text{IN}}(-)$  is  $4 \cdot 1/2$  clock periods. The maximum error voltage due to this

slight time difference between the input voltage samples is given by:

$$\Delta V_e(\text{MAX}) = (V_p) (2\pi f_{\text{cm}}) \left( \frac{4.5}{f_{\text{CLK}}} \right)$$

where:

$\Delta V_e$  is the error voltage due to sampling delay

$V_p$  is the peak value of the common-mode voltage

$f_{\text{cm}}$  is the common-mode frequency

As an example, to keep this error to  $1/4$  LSB ( $\sim 5$  mV) when operating with a 60 Hz common-mode frequency,  $f_{\text{cm}}$ , and using a 640 kHz A/D clock,  $f_{\text{CLK}}$ , would allow a peak value of the common-mode voltage,  $V_p$ , which is given by:

$$V_p = \frac{[\Delta V_e(\text{MAX})] (f_{\text{CLK}})}{(2\pi f_{\text{cm}}) (4.5)}$$

or

$$V_p = \frac{(5 \times 10^{-3}) (640 \times 10^3)}{(6.28) (60) (4.5)}$$

which gives

$$V_p \approx 1.9\text{V.}$$

The allowed range of analog input voltages usually places more severe restrictions on input common-mode noise levels.

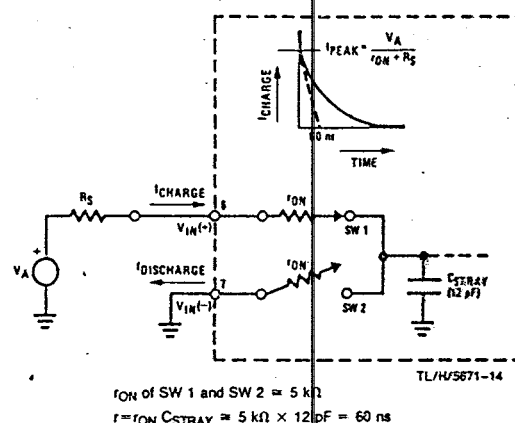
An analog input voltage with a reduced span and a relatively large zero offset can be easily handled by making use of the differential input (see section 2.4 Reference Voltage).

## 2.3 Analog Inputs

### 2.3.1 Input Current

#### Normal Mode

Due to the internal switching action, displacement currents will flow at the analog inputs. This is due to on-chip stray capacitance to ground as shown in Figure 3.



$$r_{\text{ON}} \text{ of SW 1 and SW 2} = 5 \text{ k}\Omega$$

$$t = r_{\text{ON}} C_{\text{STRAY}} = 5 \text{ k}\Omega \times 0.2 \text{ pF} = 60 \text{ ns}$$

FIGURE 3. Analog Input Impedance.



The voltage on this capacitance is switched and will result in currents entering the  $V_{IN}(+)$  input pin and leaving the  $V_{IN}(-)$  input which will depend on the analog differential input voltage levels. These current transients occur at the leading edge of the internal clocks. They rapidly decay and *do not cause errors* as the on-chip comparator is strobed at the end of the clock period.

#### Fault Mode

If the voltage source which is applied to the  $V_{IN}(+)$  pin exceeds the allowed operating range of  $V_{CC} + 50$  mV, large input currents can flow through a parasitic diode to the  $V_{CC}$  pin. If these currents could exceed the 1 mA max allowed spec, an external diode (1N914) should be added to bypass this current to the  $V_{CC}$  pin (with the current bypassed with this diode, the voltage at the  $V_{IN}(-)$  pin can exceed the  $V_{CC}$  voltage by the forward voltage of this diode).

#### 2.3.2 Input Bypass Capacitors

Bypass capacitors at the inputs will average these charges and cause a DC current to flow through the output resistances of the analog signal sources. This charge pumping action is worse for continuous conversions with the  $V_{IN}(+)$  input voltage at full-scale. For continuous conversions with a 640 kHz clock frequency with the  $V_{IN}(+)$  input at 5V, this DC current is at a maximum of approximately 5  $\mu$ A. Therefore, *bypass capacitors should not be used at the analog inputs or the  $V_{REF}/2$  pin for high resistance sources ( $> 1$  k $\Omega$ )*. If input bypass capacitors are necessary for noise filtering and high source resistance is desirable to minimize capacitor size, the detrimental effects of the voltage drop across this input resistance, which is due to the average value of the input current, can be eliminated with a full-scale adjustment while the given source resistor and input bypass capacitor are both in place. This is possible because the average value of the input current is a precise linear function of the differential input voltage.

#### 2.3.3 Input Source Resistance

Large values of source resistance where an input bypass capacitor is not used, *will not cause errors* as the input currents settle out prior to the comparison time. If a low pass filter is required in the system, use a low valued series resistor ( $\leq 1$  k $\Omega$ ) for a passive RC section or add an op amp RC active low pass filter. For low source resistance applications, ( $\leq 1$  k $\Omega$ ), a 0.1  $\mu$ F bypass capacitor at the inputs will prevent pickup due to series lead inductance of a long wire. A 100 $\Omega$  series resistor can be used to isolate this capacitor—both the R and C are placed outside the feedback loop—from the output of an op amp, if used.

#### 2.3.4 Noise

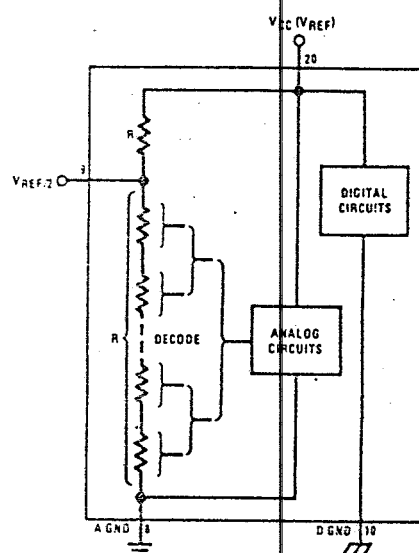
The leads to the analog inputs (pin 6 and 7) should be kept as short as possible to minimize input noise coupling. Both noise and undesired digital clock coupling to these inputs can cause system errors. The source resistance for these inputs should, in general, be kept below 5 k $\Omega$ . Larger values of source resistance can cause undesired system noise pickup. Input bypass capacitors, placed from the analog inputs to ground, will eliminate system noise pickup but can create analog scale errors as these capacitors will average the transient input switching currents of the A/D (see section 2.3.1.). This scale error depends on both a large source

resistance and the use of an input bypass capacitor. This error can be eliminated by doing a full-scale adjustment of the A/D (adjust  $V_{REF}/2$  for a proper full-scale reading—see section 2.5.2 on Full-Scale Adjustment) with the source resistance and input bypass capacitor in place.

#### 2.4 Reference Voltage

##### 2.4.1 Span Adjust

For maximum applications flexibility, these A/Ds have been designed to accommodate a 5  $V_{CC}$ , 2.5  $V_{CC}$  or an adjusted voltage reference. This has been achieved in the design of the IC as shown in *Figure 4*.



TL/H/5671-15

FIGURE 4. The  $V_{REF}$  Design on the IC

Notice that the reference voltage for the IC is either  $1/2$  of the voltage which is applied to the  $V_{CC}$  supply pin, or is equal to the voltage which is externally forced at the  $V_{REF}/2$  pin. This allows for a ratiometric voltage reference using the  $V_{CC}$  supply, a 5  $V_{CC}$  reference voltage can be used for the  $V_{CC}$  supply or a voltage less than 2.5  $V_{CC}$  can be applied to the  $V_{REF}/2$  input for increased application flexibility. The internal gain to the  $V_{REF}/2$  input is 2 making the full-scale differential input voltage twice the voltage at pin 9.

An example of the use of an adjusted reference voltage is to accommodate a reduced span—or dynamic voltage range of the analog input voltage. If the analog input voltage were to range from 0.5  $V_{CC}$  to 3.5  $V_{CC}$ , instead of 0V to 5  $V_{CC}$ , the span would be 3V as shown in *Figure 5*. With 0.5  $V_{CC}$  applied to the  $V_{IN}(-)$  pin to absorb the offset, the reference voltage can be made equal to  $1/2$  of the 3V span or 1.5  $V_{CC}$ . The A/D now will encode the  $V_{IN}(+)$  signal from 0.5V to 3.5 V with the 0.5V input corresponding to zero and the 3.5  $V_{CC}$  input corresponding to full-scale. The full 8 bits of resolution are therefore applied over this reduced analog input voltage range.

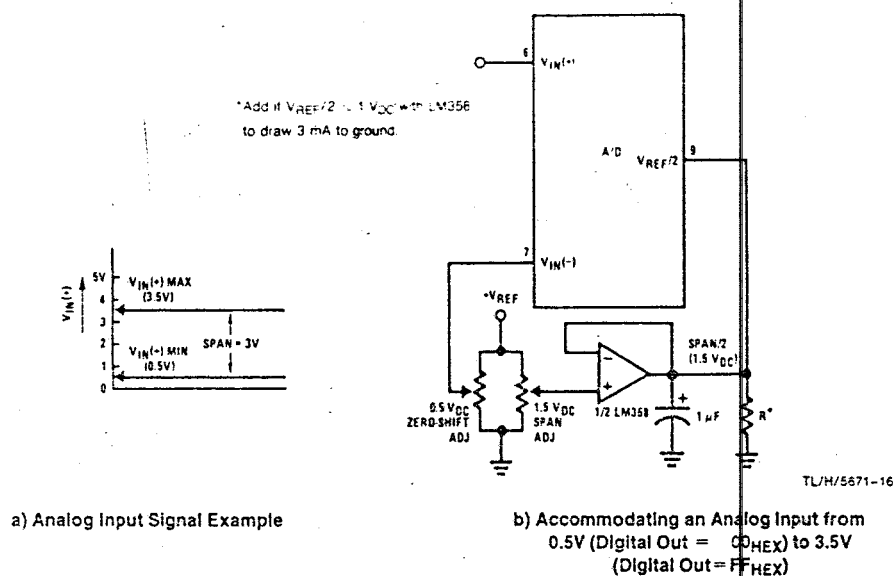


FIGURE 5. Adapting the A/D Analog Input Voltages to Match an Arbitrary Input Signal Range

#### 2.4.2 Reference Accuracy Requirements

The converter can be operated in a ratiometric mode or an absolute mode. In ratiometric converter applications, the magnitude of the reference voltage is a factor in both the output of the source transducer and the output of the A/D converter and therefore cancels out in the final digital output code. The ADC0805 is specified particularly for use in ratiometric applications with no adjustments required. In absolute conversion applications, both the initial value and the temperature stability of the reference voltage are important accuracy factors in the operation of the A/D converter. For  $V_{REF}/2$  voltages of  $2.4 V_{DC}$  nominal value, initial errors of  $\pm 10 mV_{DC}$  will cause conversion errors of  $\pm 1$  LSB due to the gain of 2 of the  $V_{REF}/2$  input. In reduced span applications, the initial value and the stability of the  $V_{REF}/2$  input voltage become even more important. For example, if the span is reduced to 2.5V, the analog input LSB voltage value is correspondingly reduced from 20 mV (5V span) to 10 mV and 1 LSB at the  $V_{REF}/2$  input becomes 5 mV. As can be seen, this reduces the allowed initial tolerance of the reference voltage and requires correspondingly less absolute change with temperature variations. Note that spans smaller than 2.5V place even tighter requirements on the initial accuracy and stability of the reference source.

In general, the magnitude of the reference voltage will require an initial adjustment. Errors due to an improper value of reference voltage appear as full-scale errors in the A/D transfer function. IC voltage regulators may be used for references if the ambient temperature changes are not excessive. The LM336B 2.5V IC reference diode (from National Semiconductor) is available which has a temperature stability of 1.8 mV typ (6 mV max) over  $0^{\circ}C \leq T_A \leq +70^{\circ}C$ . Other temperature range parts are also available.

#### 2.5 Errors and Reference Voltage Adjustments

##### 2.5.1 Zero Error

The zero of the A/D does not require adjustment. If the minimum analog input voltage value  $V_{IN(MIN)}$ , is not ground, a zero offset can be done. The converter can be made to output 0000 0000 digital code for this minimum input voltage by biasing the A/D  $V_{IN}(-)$  input at this  $V_{IN(MIN)}$  value (see Applications section). This utilizes the differential mode operation of the A/D.

The zero error of the A/D converter relates to the location of the first riser of the transfer function and can be measured by grounding the  $V_{IN}(-)$  input and applying a small magnitude positive voltage to the  $V_{IN}(+)$  input. Zero error is the difference between the actual DC input voltage which is necessary to just cause an output digital code transition from 0000 0000 to 0000 0001 and the ideal  $1/2$  LSB value ( $1/2$  LSB = 9.8 mV for  $V_{REF}/2 = 2.500 V_{DC}$ ).

##### 2.5.2 Full-Scale

The full-scale adjustment can be made by applying a differential input voltage which is  $1/2$  LSB down from the desired analog full-scale voltage range and then adjusting the magnitude of the  $V_{REF}/2$  input (pin 9 or the  $V_{CC}$  supply if pin 9 is not used) for a digital output code which is just changing from 1111 1110 to 1111 1111.

### 2.5.3 Adjusting for an Arbitrary Analog Input Voltage Range

If the analog zero voltage of the A/D is shifted away from ground (for example, to accommodate an analog input signal which does not go to ground) this new zero reference should be properly adjusted first. A  $V_{IN}(-)$  voltage which equals this desired zero reference plus  $1/2$  LSB (where the LSB is calculated for the desired analog span,  $1 \text{ LSB} = \text{analog span}/256$ ) is applied to pin 6 and the zero reference voltage at pin 7 should then be adjusted to just obtain the 00<sub>HEX</sub> to 01<sub>HEX</sub> code transition.

The full-scale adjustment should then be made (with the proper  $V_{IN}(-)$  voltage applied) by forcing a voltage to the  $V_{IN}(-)$  input which is given by:

$$V_{IN}(-) \text{ fs adj} = V_{MAX} - 1.5 \left[ \frac{(V_{MAX} - V_{MIN})}{256} \right],$$

where:

$V_{MAX}$  = The high end of the analog input range and

$V_{MIN}$  = the low end (the offset zero) of the analog range. (Both are ground referenced.)

The  $V_{REF}/2$  (or  $V_{CC}$ ) voltage is then adjusted to provide a code change from FE<sub>HEX</sub> to FF<sub>HEX</sub>. This completes the adjustment procedure.

### 2.6 Clocking Option

The clock for the A/D can be derived from the CPU clock or an external RC can be added to provide self-clocking. The CLK-IN (pin 4) makes use of a Schmitt trigger as shown in Figure 6.

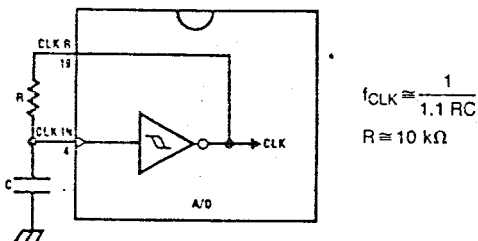


FIGURE 6. Self-Clocking the A/D

Heavy capacitive or DC loading of the clock R pin should be avoided as this will disturb normal converter operation. Loads less than 50 pF, such as driving up to 7 A/D converter clock inputs from a single clock R pin of 1 converter, are allowed. For larger clock line loading, a CMOS or low power T<sub>2</sub>L buffer or PNP input logic should be used to minimize the loading on the clock R pin (do not use a standard T<sub>2</sub>L buffer).

### 2.7 Restart During a Conversion

If the A/D is restarted ( $\overline{CS}$  and  $\overline{WR}$  go low and return high) during a conversion, the converter is reset and a new conversion is started. The output data latch is not updated if the conversion in process is not allowed to be completed, therefore the data of the previous conversion remains in this latch. The  $\overline{INTR}$  output also simply remains at the "1" level.

### 2.8 Continuous Conversions

For operation in the free-running mode an initializing pulse should be used, following power-up, to insure circuit operation. In this application, the  $\overline{CS}$  input is grounded and the  $\overline{WR}$  input is tied to the  $\overline{INTR}$  output. This  $\overline{WR}$  and  $\overline{INTR}$  node should be momentarily forced to logic low following a power-up cycle to guarantee operation.

### 2.9 Driving the Data Bus

This MOS A/D, like MOS microprocessors and memories, will require a bus driver when the total capacitance of the data bus gets large. Other circuitry which is tied to the data bus, will add to the total capacitive loading, even in TRI-STATE (high impedance mode). Backplane bussing also greatly adds to the stray capacitance of the data bus.

There are some alternatives available to the designer to handle this problem. Basically, the capacitive loading of the data bus slows down the response time, even though DC specifications are still met. For systems operating with a relatively slow CPU clock frequency, more time is available in which to establish proper logic levels on the bus and therefore higher capacitive loads can be driven (see typical characteristics curves).

At higher CPU clock frequencies time can be extended for I/O reads (and/or writes) by inserting wait states (8080) or using clock extending circuits (6800).

Finally, if time is short and capacitive loading is high, external bus drivers must be used. These can be TRI-STATE buffers (low power Schottky is recommended such as the DM74LS240 series) or special higher drive current products which are designed as bus drivers. High current bipolar bus drivers with PNP inputs are recommended.

### 2.10 Power Supplies

Noise spikes on the  $V_{CC}$  supply line can cause conversion errors as the comparator will respond to this noise. A low inductance tantalum filter capacitor should be used close to the converter  $V_{CC}$  pin and values of 1  $\mu$ F or greater are recommended. If an unregulated voltage is available in the system, a separate LM340LAZ-5.0, TO-92, 5V voltage regulator for the converter (and other analog circuitry) will greatly reduce digital noise on the  $V_{CC}$  supply.

### 2.11 Wiring and Hook-Up Precautions

Standard digital wire wrap sockets are not satisfactory for breadboarding this A/D converter. Sockets on PC boards can be used and all logic signal wires and leads should be grouped and kept as far away as possible from the analog signal leads. Exposed leads to the analog inputs can cause undesired digital noise and hum pickup, therefore shielded leads may be necessary in many applications.

A single point analog ground should be used which is separate from the logic ground points. The power supply bypass capacitor and the self-clocking capacitor (if used) should both be returned to digital ground. Any  $V_{REF}/2$  bypass capacitors, analog input filter capacitors, or input signal shielding should be returned to the analog ground point. A test for proper grounding is to measure the zero error of the A/D converter. Zero errors in excess of  $1/4$  LSB can usually be traced to improper board layout and wiring (see section 2.5.1 for measuring the zero error).

### 3.0 TESTING THE A/D CONVERTER

There are many degrees of complexity associated with testing an A/D converter. One of the simplest tests is to apply a known analog input voltage to the converter and use LEDs to display the resulting digital output code as shown in Figure 7.

For ease of testing, the  $V_{REF}/2$  (pin 9) should be supplied with 2.560 V<sub>DC</sub> and a V<sub>CC</sub> supply voltage of 5.12 V<sub>DC</sub> should be used. This provides an LSB value of 20 mV.

If a full-scale adjustment is to be made, an analog input voltage of 5.090 V<sub>DC</sub> ( $5.120 - 1/4$  LSB) should be applied to the  $V_{IN}(+)$  pin with the  $V_{IN}(-)$  pin grounded. The value of the  $V_{REF}/2$  input voltage should then be adjusted until the digital output code is just changing from 1111.1110 to 1111.1111. This value of  $V_{REF}/2$  should then be used for all the tests.

The digital output LED display can be decoded by dividing the 8 bits into 2 hex characters, the 4 most significant (MS) and the 4 least significant (LS). Table I shows the fractional binary equivalent of these two 4-bit groups. By adding the decoded voltages which are obtained from the column: Input voltage value for a 2.560 V<sub>REF/2</sub> of both the MS and the LS groups, the value of the digital display can be determined. For example, for an output LED display of 1011 0110

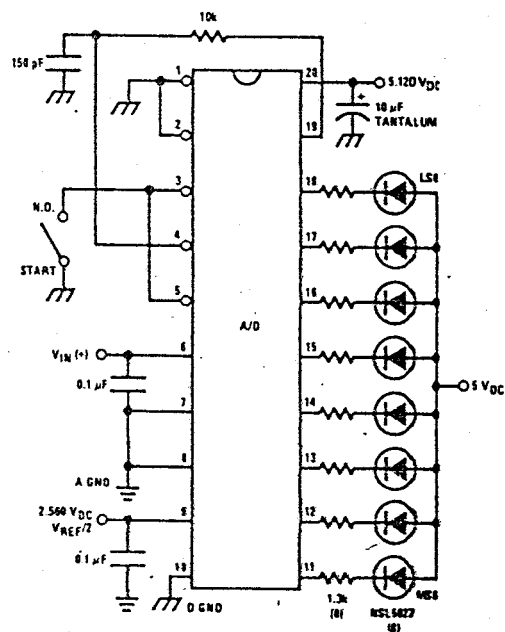


FIGURE 7. Basic A/D Tester

or B6 (in hex), the voltage values from the table are 3.520 - 0.120 or 3.640 V<sub>DC</sub>. These voltage values represent the center-values of a perfect A/D converter. The effects of quantization error have to be accounted for in the interpretation of the test results.

For a higher speed test system, or to obtain plotted data, a digital-to-analog converter is needed for the test set-up. An accurate 10-bit DAC can serve as the precision voltage source for the A/D. Errors of the A/D under test can be provided as either analog voltages or differences in 2 digital words.

A basic A/D tester which uses a DAC and provides the error as an analog output voltage is shown in Figure 8. The 2 op amps can be eliminated if a lab DVM with a numerical subtraction feature is available to directly readout the difference voltage, "A-C". The analog input voltage can be supplied by a low frequency ramp generator and an X-Y plotter can be used to provide analog error (Y axis) versus analog input (X axis). The construction details of a tester of this type are provided in the NSC application note AN-179, "Analog-to-Digital Converter Testing".

For operation with a microprocessor or a computer-based test system, it is more convenient to present the errors digitally. This can be done with the circuit of Figure 9, where the output code transitions can be detected as the 10-bit DAC is incremented. This provides  $1/4$  LSB steps for the 8-bit A/D under test. If the results of this test are automatically plotted with the analog input on the X axis and the error (in LSB's) as the Y axis, a useful transfer function of the A/D under test results. For acceptance testing, the plot is not necessary and the testing speed can be increased by establishing internal limits on the allowed error for each code.

### 4.0 MICROPROCESSOR INTERFACING

To discuss the interface with 8080A and 6800 microprocessors, a common sample subroutine structure is used. The microprocessor starts the A/D, reads and stores the results of 16 successive conversions, then returns to the user's program. The 16 data bytes are stored in 16 successive memory locations. All Data and Addresses will be given in hexadecimal form. Software and hardware details are provided separately for each type of microprocessor.

#### 4.1 Interfacing 8080 Microprocessor Derivatives (8048, 8085)

This converter has been designed to directly interface with derivatives of the 8080 microprocessor. The A/D can be mapped into memory space (using standard memory address decoding for  $\overline{CS}$  and the  $\overline{MEMR}$  and  $\overline{MEMW}$  strobes) or it can be controlled as an I/O device by using the  $\overline{I/O R}$  and  $\overline{I/O W}$  strobes and decoding the address bits A0  $\rightarrow$  A7 (or address bits A8  $\rightarrow$  A15 as they will contain the same 8-bit address information) to obtain the  $\overline{CS}$  input. Using the I/O space provides 256 additional addresses and may allow a simpler 8-bit address decoder but the data can only be input to the accumulator. To make use of the additional memory reference instructions, the A/D should be mapped into memory space. An example of an A/D in I/O space is shown in Figure 10.

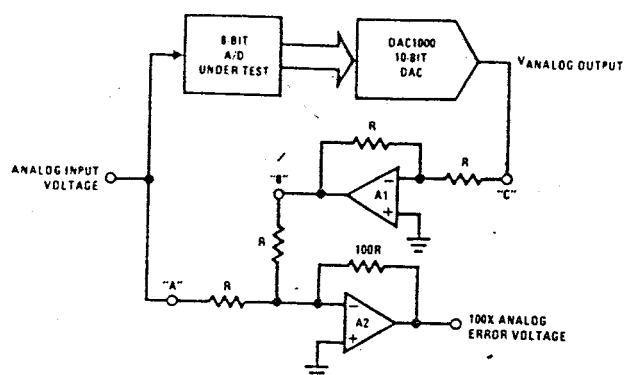


FIGURE 8. A/D Tester with Analog Error Output

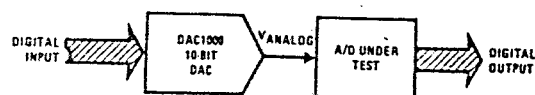


FIGURE 9. Basic "Digital" A/D Tester

TL/H/5671-19

TABLE I. DECODING THE DIGITAL OUTPUT LEDs

HEX	BINARY	FRACTIONAL BINARY VALUE FOR		OUTPUT VOLTAGE CENTER VALUES WITH $V_{REF}/2 = 2.560 V_{DC}$	
		MS GROUP	LS GROUP	VMS GROUP*	VLS GROUP*
F	1 1 1 1	15/16	15/256	4.800	0.300
E	1 1 1 0	7/8	7/128	4.480	0.280
D	1 1 0 1	13/16	13/256	4.160	0.260
C	1 1 0 0	3/4	3/64	3.840	0.240
B	1 0 1 1	11/16	11/256	3.520	0.220
A	1 0 1 0	5/8	5/128	3.200	0.200
9	1 0 0 1	9/16	9/256	2.880	0.180
8	1 0 0 0	1/2	1/32	2.560	0.160
7	0 1 1 1	7/16	7/256	2.240	0.140
6	0 1 1 0	3/8	3/128	1.920	0.120
5	0 1 0 1	5/16	2/256	1.600	0.100
4	0 1 0 0	1/4	1/64	1/280	0.080
3	0 0 1 1	3/16	3/256	0.960	0.060
2	0 0 1 0	1/8	1/128	0.640	0.040
1	0 0 0 1	1/16	1/256	0.320	0.020
0	0 0 0 0			0	0

\*Display Output = VMS Group + VLS Group



A to D, D to A

## DAC0808, DAC0807, DAC0806 8-Bit D/A Converters

### General Description

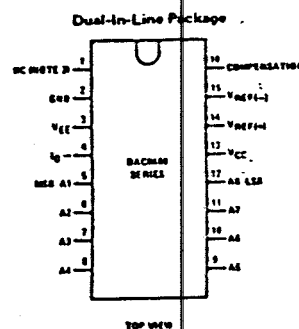
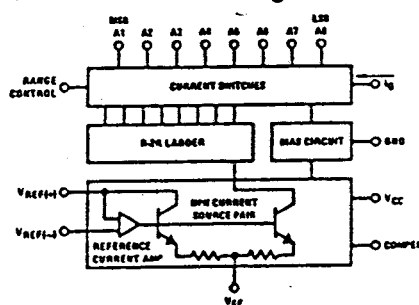
The DAC0808 series is an 8-bit monolithic digital-to-analog converter (DAC) featuring a full scale output current settling time of 150 ns while dissipating only 33 mW with  $\pm 5V$  supplies. No reference current ( $I_{REF}$ ) trimming is required for most applications since the full scale output current is typically  $\pm 1$  LSB of  $255 I_{REF}/256$ . Relative accuracies of better than  $\pm 0.19\%$  assure 8-bit monotonicity and linearity while zero level output current of less than  $4 \mu A$  provides 8-bit zero accuracy for  $I_{REF} \geq 2$  mA. The power supply currents of the DAC0808 series are independent of bit codes, and exhibits essentially constant device characteristics over the entire supply voltage range.

The DAC0808 will interface directly with popular TTL, DTL or CMOS logic levels, and is a direct replacement for the MC1508/MC1408. For higher speed applications, see DAC0800 data sheet.

### Features

- Relative accuracy:  $\pm 0.19\%$  error maximum (DAC0808)
- Full scale current match:  $\pm 1$  LSB typ
- 7 and 6-bit accuracy available (DAC0807, DAC0806)
- Fast settling time: 150 ns typ
- Noninverting digital inputs are TTL and CMOS compatible
- High speed multiplying input slew rate: 8 mA/ $\mu s$
- Power supply voltage range:  $\pm 4.5V$  to  $\pm 18V$
- Low power consumption: 33 mW @  $\pm 5V$

### Block and Connection Diagrams



### Typical Application

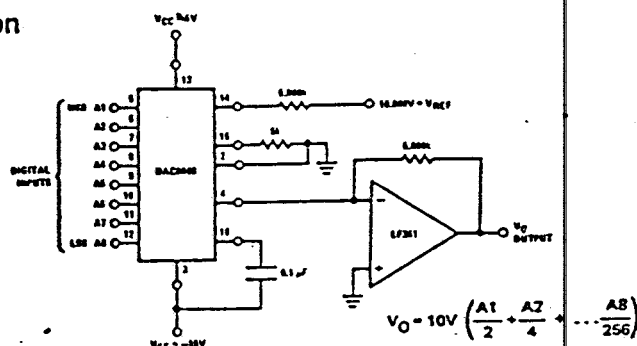


FIGURE 1. +10V Output Digital to Analog Converter

### Ordering Information

ACCURACY	OPERATING TEMPERATURE RANGE	ORDER NUMBERS*			
		D PACKAGE (D16C)		J PACKAGE (J16A)	
8-bit	$-55^{\circ}C \leq T_A \leq +125^{\circ}C$	DAC0808LD	MC1508L8	DAC0808LCJ	MC1408L8
8-bit	$0^{\circ}C \leq T_A \leq +75^{\circ}C$			DAC0807LCJ	MC1408L7
7-bit	$0^{\circ}C \leq T_A \leq +75^{\circ}C$			DAC0806LCJ	MC1408L6
6-bit	$0^{\circ}C \leq T_A \leq +75^{\circ}C$				

\*Note: Devices may be ordered by using either order number.

# Absolute Maximum Ratings

Supply Voltage	+18 V <sub>DC</sub>	Power Dissipation (Package Limitation)	1000 mW
V <sub>CC</sub>	+18 V <sub>DC</sub>	Derate above T <sub>A</sub> = 25°C	6.7 mW/°C
V <sub>EE</sub>	-18 V <sub>DC</sub>	Operating Temperature Range	
Digital Input Voltage, V <sub>I</sub>	-10 V <sub>DC</sub> to +18 V <sub>DC</sub>	DAC0808L	-55°C ≤ T <sub>A</sub> ≤ +125°C
Digital Output Voltage, V <sub>O</sub>	-11 V <sub>DC</sub> to +18 V <sub>DC</sub>	DAC0808LC Series	0 ≤ T <sub>A</sub> ≤ +75°C
Reference Current, I <sub>14</sub>	5 mA	Storage Temperature Range	-65°C to +150°C
Reference Amplifier Inputs, V <sub>14</sub> , V <sub>15</sub>	V <sub>CC</sub> , V <sub>EE</sub>		

## Electrical Characteristics

V<sub>CC</sub> = 5V, V<sub>EE</sub> = -15V<sub>DC</sub>, V<sub>REF</sub>/R<sub>14</sub> = 2 mA, DAC0808: T<sub>A</sub> = -55°C to +125°C, DAC0808C, DAC0807C, DAC0806C, T<sub>A</sub> = 0°C to +75°C, and all digital inputs at high logic level unless otherwise noted.)

PARAMETER	CONDITIONS	MIN	TYP	MAX	UNITS
Relative Accuracy (Error Relative to Full Scale I <sub>Q</sub> )	(Figure 4)				%
DAC0808L (LM1508-8)				±0.19	%
DAC0808LC (LM1408-8)				±0.39	%
DAC0807LC (LM1408-7), (Note 1)				±0.78	%
DAC0806LC (LM1408-6), (Note 1)					%
Settling Time to Within 1/2 LSB (Includes t <sub>PLH</sub> )	T <sub>A</sub> = 25°C (Note 2), (Figure 5)		150		ns
Propagation Delay Time	T <sub>A</sub> = 25°C, (Figure 5)		30	100	ns
Output Full Scale Current Drift			±20		ppm/°C
Digital Input Logic Levels	(Figure 3)				V <sub>DC</sub>
High Level, Logic "1"		2		0.8	V <sub>DC</sub>
Low Level, Logic "0"					V <sub>DC</sub>
Digital Input Current	(Figure 3)				mA
High Level	V <sub>IH</sub> = 5V		0	0.040	mA
Low Level	V <sub>IL</sub> = 0.8V		-0.003	-0.8	mA
Reference Input Bias Current	(Figure 3)		-1	-3	μA
Output Current Range	(Figure 3)				mA
	V <sub>EE</sub> = -5V	0	2.0	2.1	mA
	V <sub>EE</sub> = -15V, T <sub>A</sub> = 25°C	0	2.0	4.2	mA
Output Current	V <sub>REF</sub> = 2.000V, R <sub>14</sub> = 1000Ω, (Figure 3)	1.9	1.99	2.1	mA
Output Current, All Bits Low	(Figure 3)		0	4	μA
Output Voltage Compliance	E <sub>r</sub> ≤ 0.19%, T <sub>A</sub> = 25°C			-0.55, +0.4	V <sub>DC</sub>
P <sub>m</sub> 1 Grounded, V <sub>EE</sub> Below -10V				-5.0, +0.4	V <sub>DC</sub>
Reference Current Slew Rate	(Figure 6)	4	8		mA/μs
Output Current Power Supply Sensitivity	-5V ≤ V <sub>EE</sub> ≤ -15V		0.05	2.7	μA/V
Power Supply Current (All Bits Low)	(Figure 3)				mA
			2.3	22	mA
			-4.3	-13	mA
Power Supply Voltage Range	T <sub>A</sub> = 25°C, (Figure 3)	4.5	5.0	5.5	V <sub>DC</sub>
		-4.5	-15	-16.5	V <sub>DC</sub>
Power Dissipation					mW
All Bits Low	V <sub>CC</sub> = 5V, V <sub>EE</sub> = -5V		33	170	mW
	V <sub>CC</sub> = 5V, V <sub>EE</sub> = -15V		106	305	mW
All Bits High	V <sub>CC</sub> = 15V, V <sub>EE</sub> = -5V		90		mW
	V <sub>CC</sub> = 15V, V <sub>EE</sub> = -15V		160		mW

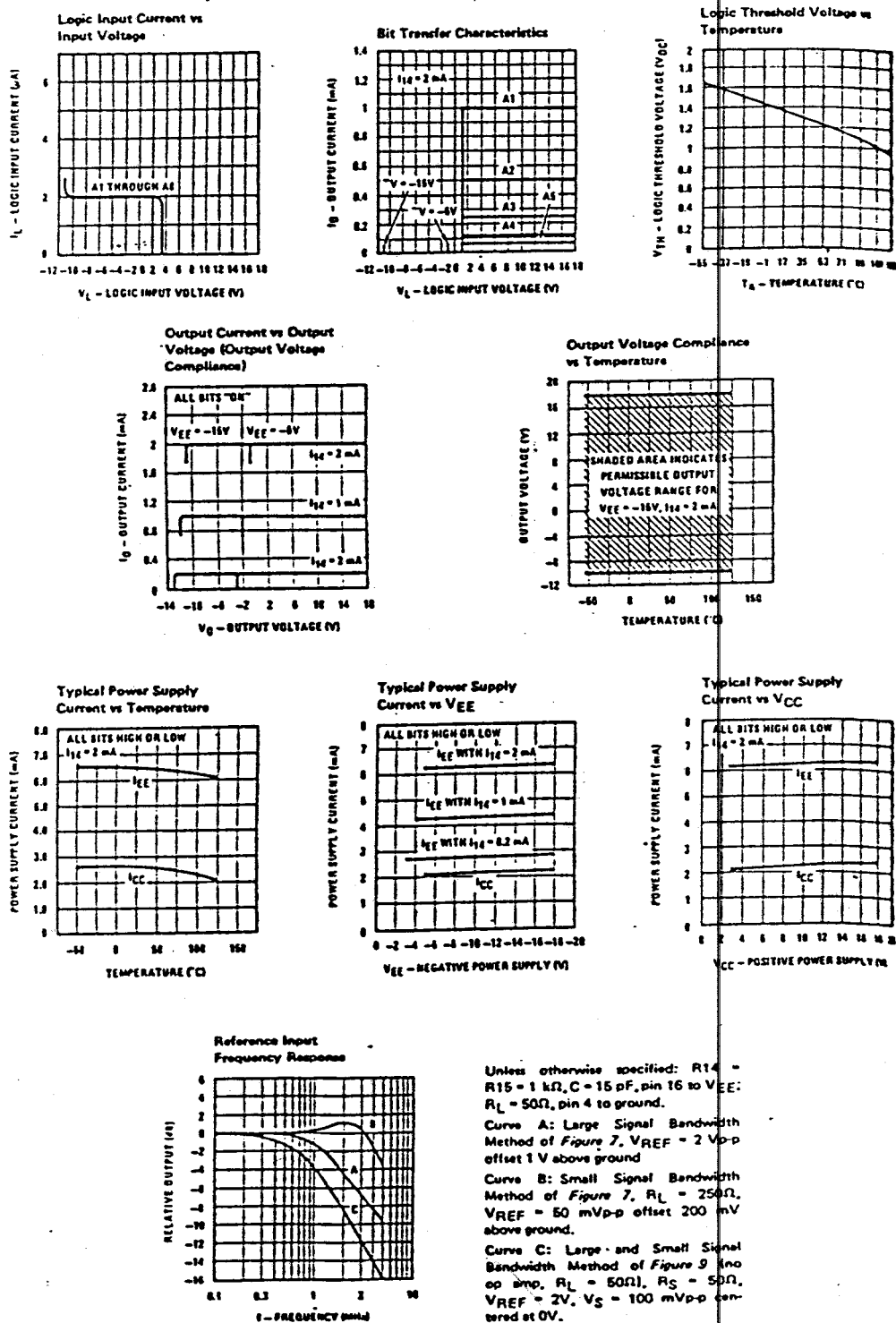
Note 1: All current switches are tested to guarantee at least 50% of rated current.

Note 2: All bits switched.

Note 3: Range control is not required.

# Typical Performance Characteristics

$V_{CC} = 5V$ ,  $V_{EE} = -15V$ ,  $T_A = 25^\circ C$ , unless otherwise noted





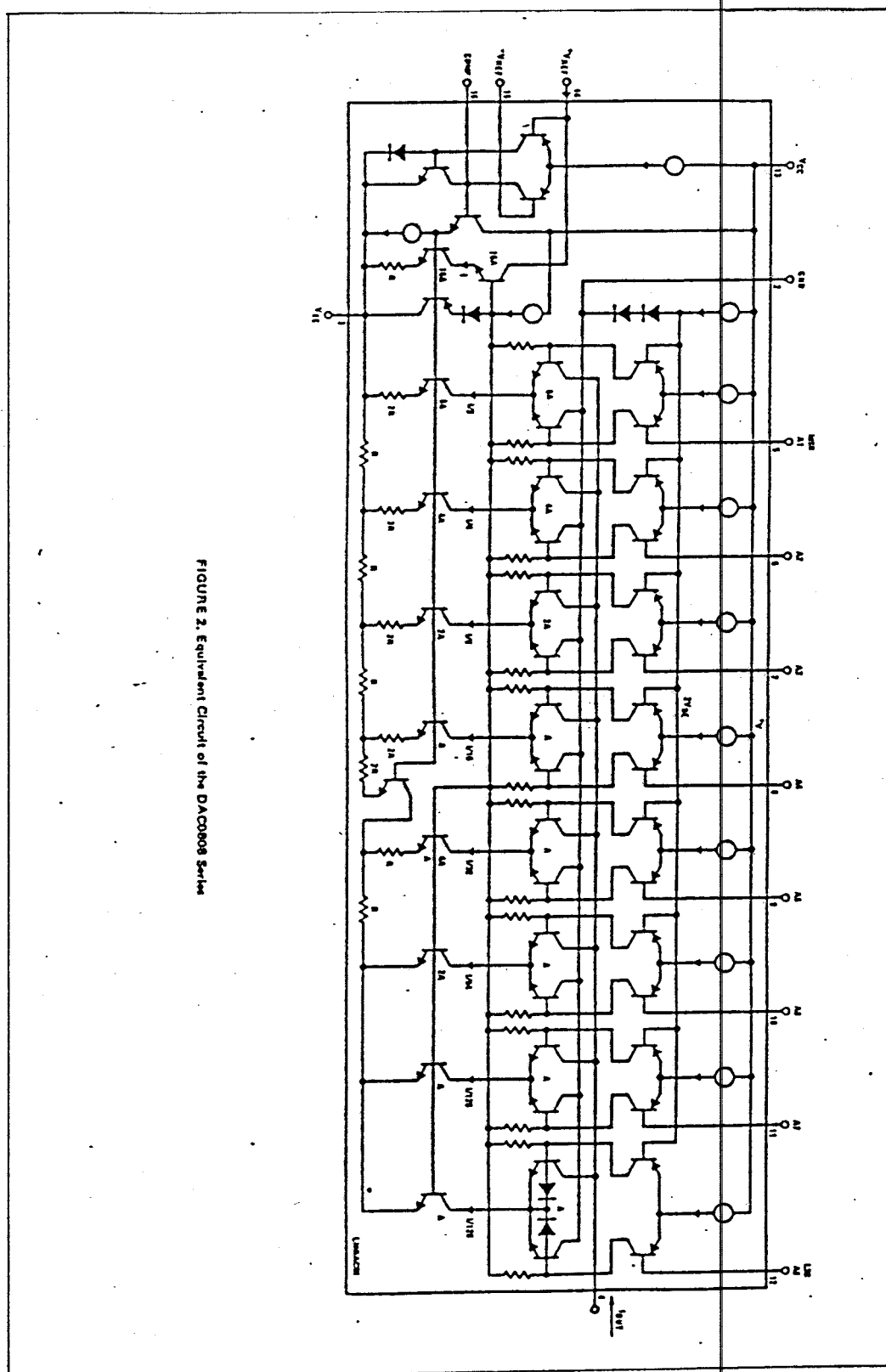


FIGURE 2. Equivalent Circuit of the DAC0808 Series



## Test Circuits (Continued)

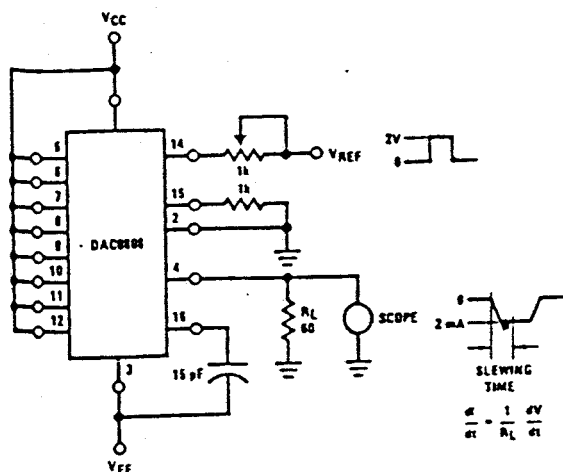


FIGURE 6. Reference Current Slow Rate Measurement

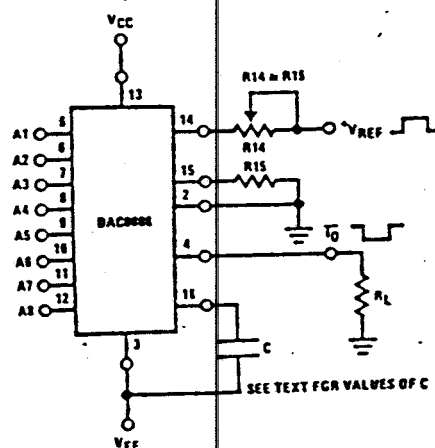


FIGURE 7. Positive VREF

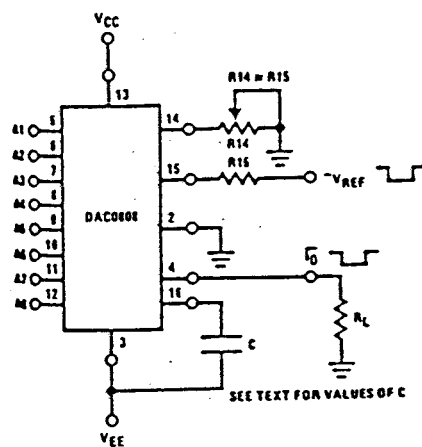


FIGURE 8. Negative VREF

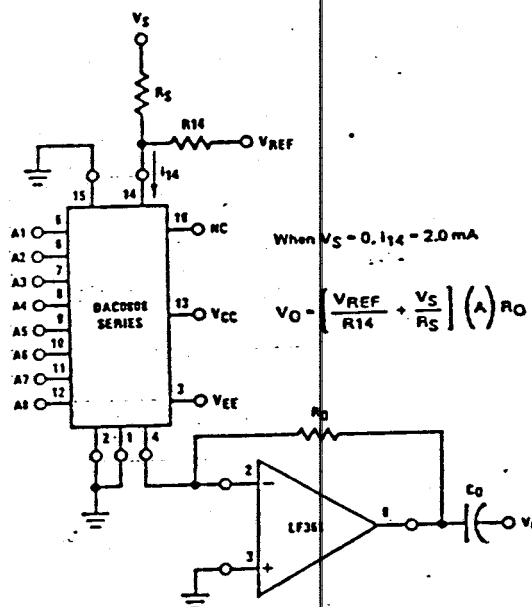


FIGURE 9. Programmable Gain Amplifier or Digital Attenuator Circuit

## Application Hints

## REFERENCE AMPLIFIER DRIVE AND COMPENSATION

The reference amplifier provides a voltage at pin 14 for converting the reference voltage to a current, and a turn-around circuit or current mirror for feeding the ladder. The reference amplifier input current,  $I_{14}$ , must always flow into pin 14, regardless of the set-up method or reference voltage polarity.

Connections for a positive voltage are shown in Figure 7. The reference voltage source supplies the full current

$I_{14}$ . For bipolar reference signals, as in the multiplying mode,  $R_{15}$  can be tied to a negative voltage corresponding to the minimum input level. It is possible to eliminate  $R_{15}$  with only a small sacrifice in accuracy and temperature drift.

The compensation capacitor value must be increased with increases in  $R_{14}$  to maintain proper phase margin; for  $R_{14}$  values of 1, 2.5 and 5 k $\Omega$ , minimum capacitor values are 15, 37 and 75 pF. The capacitor may be tied to either VEE or ground, but using VEE increases negative supply rejection.

### Application Hints (Continued)

A negative reference voltage may be used if R14 is grounded and the reference voltage is applied to R15 as shown in Figure 8. A high input impedance is the main advantage of this method. Compensation involves a capacitor to VEE on pin 16, using the values of the previous paragraph. The negative reference voltage must be at least 4V above the VEE supply. Bipolar input signals may be handled by connecting R14 to a positive reference voltage equal to the peak positive input level at pin 15.

When a DC reference voltage is used, capacitive bypass to ground is recommended. The 5V logic supply is not recommended as a reference voltage. If a well regulated 5V supply which drives logic is to be used as the reference, R14 should be decoupled by connecting it to 5V through another resistor and bypassing the junction of the 2 resistors with 0.1  $\mu$ F to ground. For reference voltages greater than 5V, a clamp diode is recommended between pin 14 and ground.

If pin 14 is driven by a high impedance such as a transistor current source, none of the above compensation methods apply and the amplifier must be heavily compensated, decreasing the overall bandwidth.

#### OUTPUT VOLTAGE RANGE

The voltage on pin 4 is restricted to a range of -0.6 to 0.5V when VEE = -5V due to the current switching methods employed in the DAC0808.

The negative output voltage compliance of the DAC0808 is extended to -5V where the negative supply voltage is more negative than -10V. Using a full-scale current of 1.992 mA and load resistor of 2.5 k $\Omega$  between pin 4 and ground will yield a voltage output of 256 levels between 0 and -4.980V. Floating pin 1 does not affect the converter speed or power dissipation. However, the value of the load resistor determines the switching time due to increased voltage swing. Values of  $R_L$  up to 500 $\Omega$  do not significantly affect performance, but a 2.5 k $\Omega$  load increases worst-case settling time to 1.2  $\mu$ s (when all bits are switched ON). Refer to the subsequent text section on Settling Time for more details on output loading.

#### OUTPUT CURRENT RANGE

The output current maximum rating of 4.2 mA may be used only for negative supply voltages more negative than -7V, due to the increased voltage drop across the resistors in the reference current amplifier.

#### ACCURACY

Absolute accuracy is the measure of each output current level with respect to its intended value, and is dependent upon relative accuracy and full-scale current drift. Relative accuracy is the measure of each output current level as a fraction of the full-scale current. The relative accuracy of the DAC0808 is essentially constant with temperature due to the excellent temperature tracking

of the monolithic resistor ladder. The reference current may drift with temperature, causing a change in the absolute accuracy of output current. However, the DAC0808 has a very low full-scale current drift with temperature.

The DAC0808 series is guaranteed accurate to within  $\pm 1/2$  LSB at a full-scale output current of 1.992 mA. This corresponds to a reference amplifier output current drive to the ladder network of 2 mA, with the loss of 1 LSB (8  $\mu$ A) which is the ladder remainder shunted to ground. The input current to pin 14 has a guaranteed value of between 1.9 and 2.1 mA, allowing some mismatch in the NPN current source pair. The accuracy test circuit is shown in Figure 4. The 12-bit converter is calibrated for a full-scale output current of 1.992 mA. This is an optional step since the DAC0808 accuracy is essentially the same between 1.5 and 2.5 mA. Then the DAC0808 circuits' full-scale current is trimmed to the same value with R14 so that a zero value appears at the error amplifier output. The counter is activated and the error band may be displayed on an oscilloscope, detected by comparators, or stored in a peak detector.

Two 8-bit D-to-A converters may not be used to construct a 16-bit accuracy D-to-A converter. 16-bit accuracy implies a total error of  $\pm 1/2$  of one part in 65,536, or  $\pm 0.00076\%$ , which is much more accurate than the  $\pm 0.019\%$  specification provided by the DAC0808.

#### MULTIPLYING ACCURACY

The DAC0808 may be used in the multiplying mode with 8-bit accuracy when the reference current is varied over a range of 256:1. If the reference current in the multiplying mode ranges from 16  $\mu$ A to 4 mA, the additional error contributions are less than 1.6  $\mu$ A. The is well within 8-bit accuracy when referred to full-scale.

A monotonic converter is one which supplies an increase in current for each increment in the binary word. Typically, the DAC0808 is monotonic for all values of reference current above 0.5 mA. The recommended range for operation with a DC reference current is 0.5 to 4 mA.

#### SETTLING TIME

The worst-case switching condition occurs when all bits are switched ON, which corresponds to a low-to-high transition for all bits. This time is typically 150 ns for settling to within  $\pm 1/2$  LSB, for 8-bit accuracy, and 100 ns to 1/2 LSB for 7 and 6-bit accuracy. The turn OFF is typically under 100 ns. These times apply when  $R_L \leq 500\Omega$  and  $C_O \leq 25$  pF.

Extra care must be taken in board layout since this is usually the dominant factor in satisfactory test results when measuring settling time. Short leads, 100  $\mu$ F supply bypassing for low frequencies, and minimum scope lead length are all mandatory.



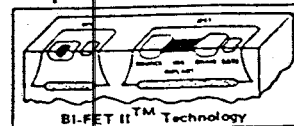
## LF351 Wide Bandwidth JFET Input Operational Amplifier

### General Description

The LF351 is a low cost high speed JFET input operational amplifier with an internally trimmed input offset voltage (BI-FET II™ technology). The device requires a low supply current and yet maintains a large gain bandwidth product and a fast slew rate. In addition, well matched high voltage JFET input devices provide very low input bias and offset currents. The LF351 is pin compatible with the standard LM741 and uses the same offset voltage adjustment circuitry. This feature allows designers to immediately upgrade the overall performance of existing LM741 designs.

The LF351 may be used in applications such as high speed integrators, fast D/A converters, sample-and-hold circuits and many other circuits requiring low input offset voltage, low input bias current, high input impedance, high slew rate and wide bandwidth. The device has low noise and offset voltage drift, but for applica-

### Operational Amplifiers/Buffers

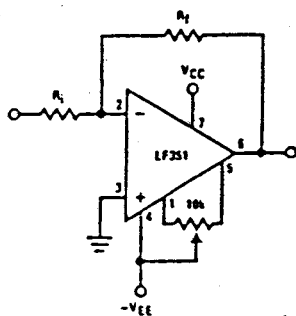


tions where these requirements are critical, the LF351 is recommended. If maximum supply current is important however, the LF351 is the better choice.

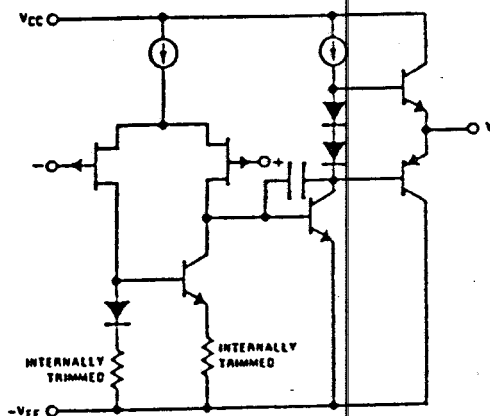
### Features

- Internally trimmed offset voltage 10 mV
- Low input bias current 50 pA
- Low input noise voltage 16 nV/√Hz
- Low input noise current 0.01 pA/√Hz
- Wide gain bandwidth 4 MHz
- High slew rate 13 V/μs
- Low supply current 1.8 mA
- High input impedance 10<sup>12</sup> Ω
- Low total harmonic distortion  $A_v = 10$ ,  $R_L = 10k$ ,  $V_O = 20$  Vp-p, BW = 20 Hz-20 kHz <0.02%
- Low 1/f noise corner 50 Hz
- Fast settling time to 0.01% 2 μs

### Typical Connection

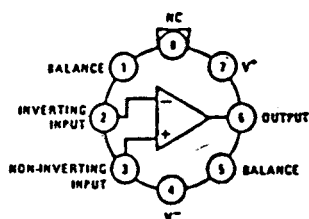


### Simplified Schematic



### Connection Diagrams (Top Views)

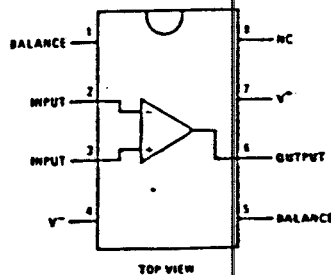
#### Metal Can Package



Note: Pin 4 connected to case.

Order Number LF351H  
See NS Package H08C -

#### Dual-In-Line Package



Order Number LF351N  
See NS Package N08A

### Absolute Maximum Ratings

Supply Voltage	$\pm 18\text{V}$
Power Dissipation (Note 1)	500mW
Operating Temperature Range	$0^{\circ}\text{C}$ to $+70^{\circ}\text{C}$
$T_J(\text{MAX})$	$115^{\circ}\text{C}$
Differential Input Voltage	$\pm 30\text{V}$
Input Voltage Range (Note 2)	$\pm 15\text{V}$
Output Short Circuit Duration	Continuous
Storage Temperature Range	$-65^{\circ}\text{C}$ to $+150^{\circ}\text{C}$
Lead Temperature (Soldering, 10 seconds)	$300^{\circ}\text{C}$

### DC Electrical Characteristics (Note 3)

SYMBOL	PARAMETER	CONDITIONS	LF351			UNITS
			MIN	TYP	MAX	
$V_{OS}$	Input Offset Voltage	$R_S = 10\text{k}\Omega$ , $T_A = 25^{\circ}\text{C}$ Over Temperature		5	10 13	mV mV
$\Delta V_{OS}/\Delta T$	Average TC of Input Offset Voltage	$R_S = 10\text{k}\Omega$		10		$\mu\text{V}/^{\circ}\text{C}$
$I_{OS}$	Input Offset Current	$T_J = 25^{\circ}\text{C}$ , (Notes 3, 4) $T_J < 70^{\circ}\text{C}$		25	100 4	pA nA
$I_B$	Input Bias Current	$T_J = 25^{\circ}\text{C}$ , (Notes 3, 4) $T_J < 70^{\circ}\text{C}$		50	200 8	pA nA
$R_{IN}$	Input Resistance	$T_J = 25^{\circ}\text{C}$		1012		$\Omega$
$A_{VOL}$	Large Signal Voltage Gain	$V_S = \pm 15\text{V}$ , $T_A = 25^{\circ}\text{C}$ $V_O = \pm 10\text{V}$ , $R_L = 2\text{k}\Omega$ Over Temperature	25	100		V/mV V/mV
$V_O$	Output Voltage Swing	$V_S = \pm 15\text{V}$ , $R_L = 10\text{k}\Omega$	$\pm 12$	$\pm 13.5$		V
$V_{CM}$	Input Common-Mode Voltage Range	$V_S = \pm 15\text{V}$	$\pm 11$	$+15$ $-12$		V V
CMRR	Common-Mode Rejection Ratio	$R_S < 10\text{k}\Omega$	70	100		dB
PSRR	Supply Voltage Rejection Ratio	(Note 5)	70	100		dB
$I_S$	Supply Current			1.8	3.4	mA

### AC Electrical Characteristics (Note 3)

SYMBOL	PARAMETER	CONDITIONS	LF351			UNITS
			MIN	TYP	MAX	
SR	Slew Rate	$V_S = \pm 15\text{V}$ , $T_A = 25^{\circ}\text{C}$		13		V/ $\mu\text{s}$
GBW	Gain Bandwidth Product	$V_S = \pm 15\text{V}$ , $T_A = 25^{\circ}\text{C}$		4		MHz
$e_n$	Equivalent Input Noise Voltage	$T_A = 25^{\circ}\text{C}$ , $R_S = 100\Omega$ , $f = 1000\text{Hz}$		16		nV/ $\sqrt{\text{Hz}}$
$i_n$	Equivalent Input Noise Current	$T_J = 25^{\circ}\text{C}$ , $f = 1000\text{Hz}$		0.01		pA/ $\sqrt{\text{Hz}}$

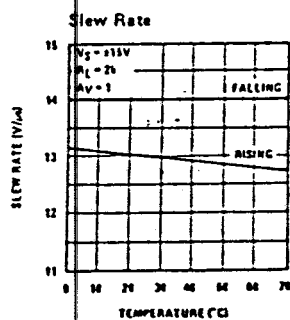
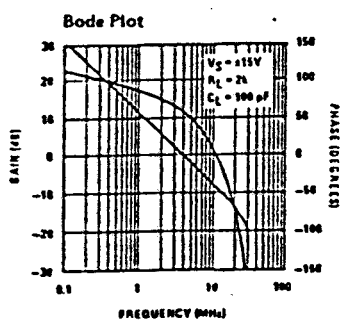
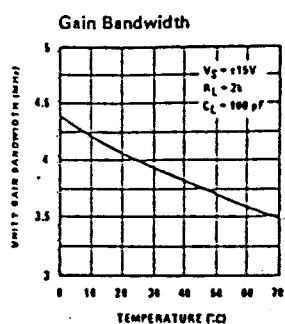
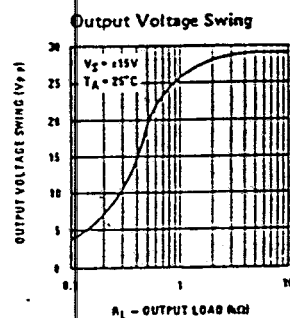
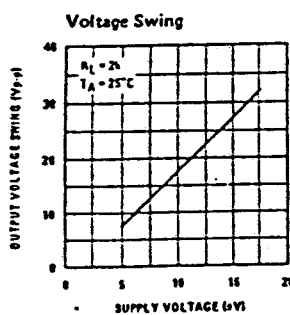
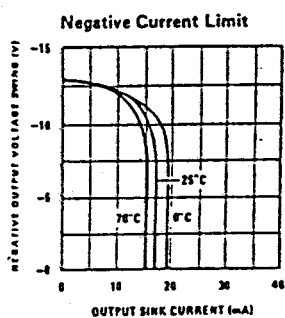
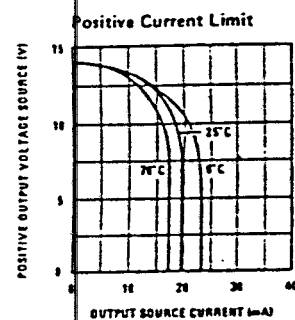
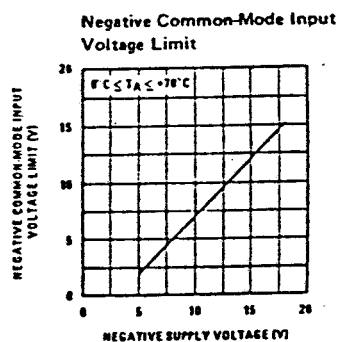
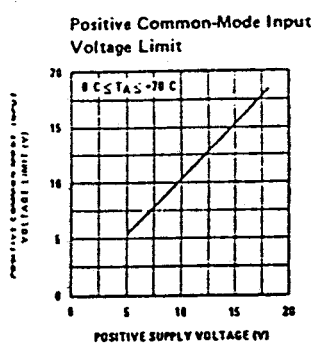
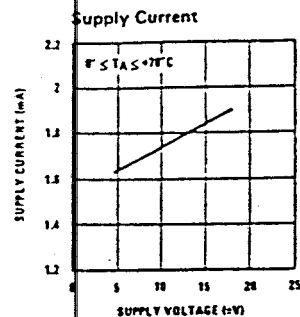
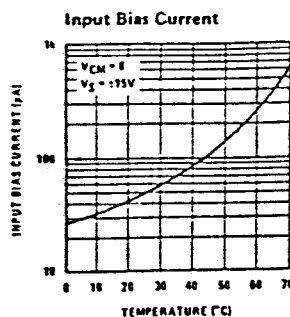
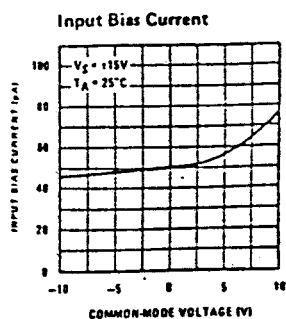
Note 1: For operating at elevated temperature, the device must be derated based on a thermal resistance of  $150^{\circ}\text{C}/\text{W}$  junction to ambient or  $45^{\circ}\text{C}/\text{W}$  junction to case.

Note 2: Unless otherwise specified the absolute maximum negative input voltage is equal to the negative power supply voltage.

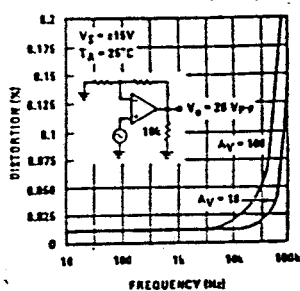
Note 3: These specifications apply for  $V_S = \pm 15\text{V}$  and  $0^{\circ}\text{C} < T_A < +70^{\circ}\text{C}$ .  $V_{OS}$ ,  $I_B$  and  $I_{OS}$  are measured at  $V_{CM} = 0$ .  
Note 4: The input bias currents are junction leakage currents which approximately double for every  $10^{\circ}\text{C}$  increase in the junction temperature,  $T_J$ . Due to the limited production test time, the input bias currents measured are correlated to junction temperature. In normal operation the junction temperature rises above the ambient temperature as a result of internal power dissipation,  $P_D$ .  $T_J = T_A + \theta_{JA} P_D$  where  $\theta_{JA}$  is the thermal resistance from junction to ambient. Use of a heat sink is recommended if input bias current is to be kept at minimum.

Note 5: Supply voltage rejection ratio is measured for both supply magnitudes increasing or decreasing simultaneously in accordance with common practice.

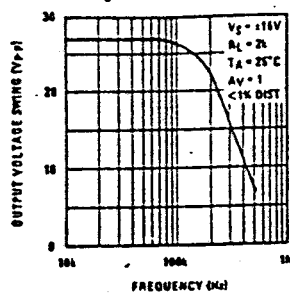
## Typical Performance Characteristics



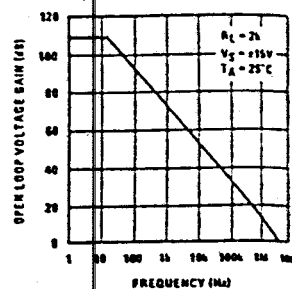
Distortion vs Frequency



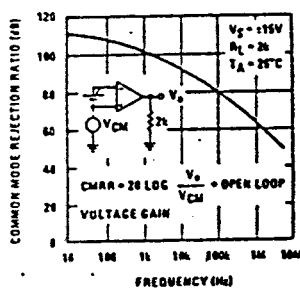
Undistorted Output Voltage Swing



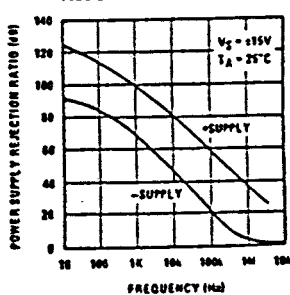
Open Loop Frequency Respo



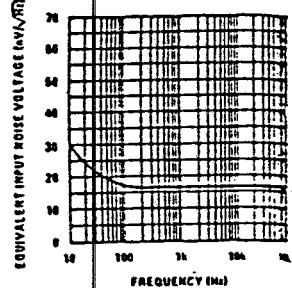
Common-Mode Rejection Ratio



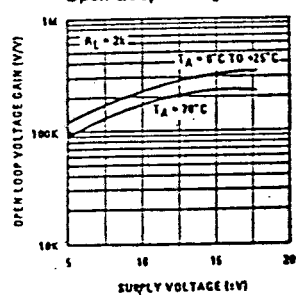
Power Supply Rejection Ratio



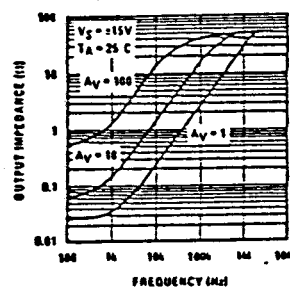
Equivalent Input Noise Voltage



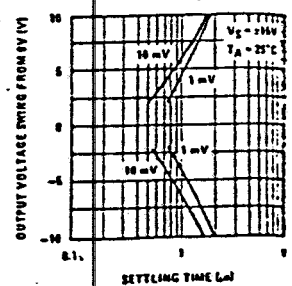
Open Loop Voltage Gain (V/V)



Output Impedance



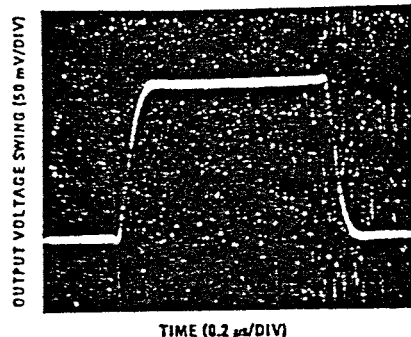
Inverter Settling Time



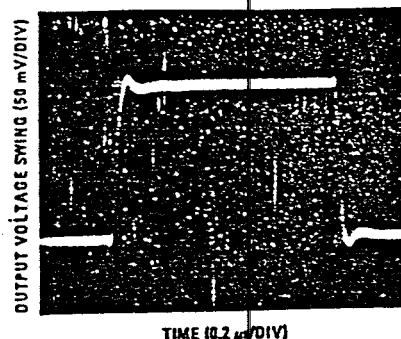


## Pulse Response

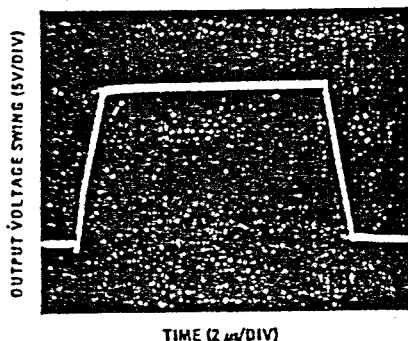
Small Signal Inverting



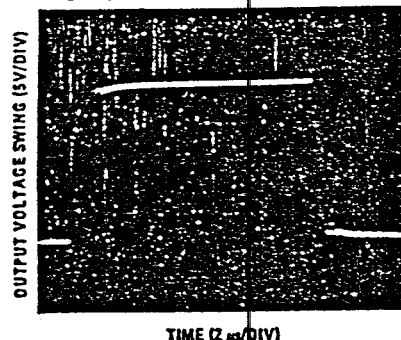
Small Signal Non-Inverting



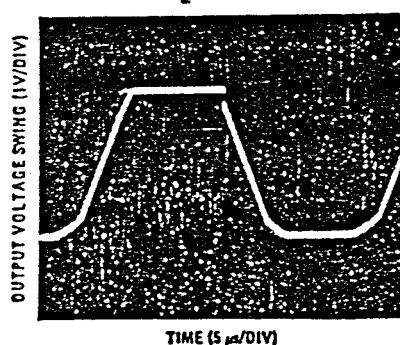
Large Signal Inverting



Large Signal Non-Inverting



Current Limit ( $R_L = 100\Omega$ )



## Application Hints

The LF351 is an op amp with an internally trimmed input offset voltage and JFET input devices (BI-FET II<sup>TM</sup>). These JFETs have large reverse breakdown voltages from gate to source and drain eliminating the need for clamps across the inputs. Therefore, large differential input voltages can easily be accommodated without a large increase in input current. The maximum differential input voltage is independent of the supply voltages. However, neither of the input voltages should be

allowed to exceed the negative supply as this will cause large currents to flow which can result in a destroyed unit.

Exceeding the negative common-mode limit on either input will cause a reversal of the phase to the output and force the amplifier output to the corresponding high or low state. Exceeding the negative common-mode limit on both inputs will force the amplifier output to a

### Application Hints (Continued)

high state. In neither case does a latch occur since raising the input back within the common-mode range again puts the input stage and thus the amplifier in a normal operating mode.

Exceeding the positive common-mode limit on a single input will not change the phase of the output; however, if both inputs exceed the limit, the output of the amplifier will be forced to a high state.

The amplifier will operate with a common-mode input voltage equal to the positive supply; however, the gain bandwidth and slew rate may be decreased in this condition. When the negative common-mode voltage swings to within 3V of the negative supply, an increase in input offset voltage may occur.

The LF351 is biased by a zener reference which allows normal circuit operation on  $\pm 4V$  power supplies. Supply voltages less than these may result in lower gain bandwidth and slew rate.

The LF351 will drive a  $2\text{ k}\Omega$  load resistance to  $\pm 10V$  over the full temperature range of  $0^\circ\text{C}$  to  $+70^\circ\text{C}$ . If the amplifier is forced to drive heavier load currents, however, an increase in input offset voltage may occur on the negative voltage swing and finally reach an active current limit on both positive and negative swings.

Precautions should be taken to ensure that the power supply for the integrated circuit never becomes reversed in polarity or that the unit is not inadvertently installed

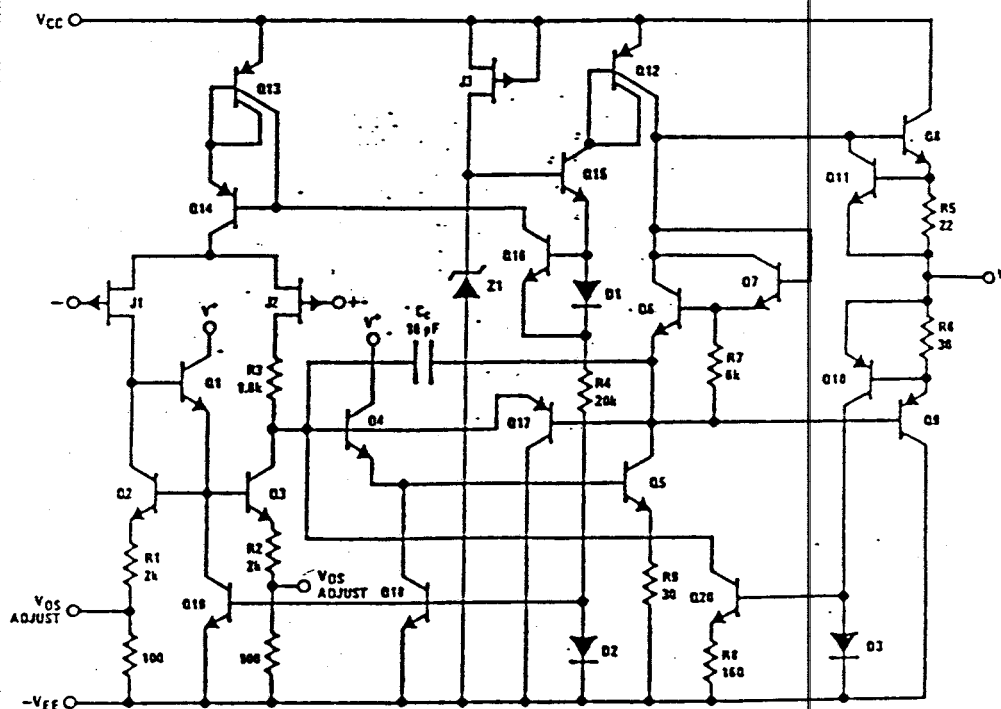
backwards in a socket as an unlimited current surge through the resulting forward diode within the IC could cause fusing of the internal conductors and result in a destroyed unit.

Because these amplifiers are JFET rather than MOSFET input op amps they do not require special handling.

As with most amplifiers, care should be taken with lead dress, component placement and supply decoupling in order to ensure stability. For example, resistors from the output to an input should be placed with the body close to the input to minimize "pick-up" and maximize the frequency of the feedback pole by minimizing the capacitance from the input to ground.

A feedback pole is created when the feedback around any amplifier is resistive. The parallel resistance and capacitance from the input of the device (usually the inverting input) to AC ground set the frequency of the pole. In many instances the frequency of this pole is much greater than the expected 3 dB frequency of the closed loop gain and consequently there is negligible effect on stability margin. However, if the feedback pole is less than approximately 6 times the expected 3 dB frequency a lead capacitor should be placed from the output to the input of the op amp. The value of the added capacitor should be such that the RC time constant of this capacitor and the resistance it parallels is greater than or equal to the original feedback pole time constant.

### Detailed Schematic



## USULAN TUGAS AKHIR

A. JUDUL : IMPLEMENTASI PROSESOR SINYAL DIGITAL  
TMS32010 UNTUK PENGATURAN POSISI  
MENGUNAKAN MOTOR DC

B. RUANG LINGKUP : - Mikroprosesor  
- Pengolahan Sinyal Digital  
- Sistem Pengaturan

C. LATAR BELAKANG : Munculnya chip-chip mikroprosesor khusus dengan kemampuan proses yang sangat cepat dibandingkan dengan mikroprosesor tipe umum, memungkinkan pengaturan suatu sistem yang pada mulanya dilakukan secara analog dapat diubah menjadi sistem pengaturan digital karena cara ini mempunyai beberapa kelebihan daripada sistem analog.

Dengan sistem pengaturan digital dalam hal ini dengan menggunakan prosesor sinyal digital, parameter-parameter yang diperlukan misalnya konstanta penguatan dapat diubah-ubah hanya dengan mengubah data pada software (misalnya melalui input keyboard) tanpa diperlukan pengaturan pada rangkaian hardware-nya.

Prosesor sinyal digital TMS32010 sebagai prosesor khusus untuk operasi pengolahan sinyal mempunyai waktu eksekusi yang sangat cepat yaitu sebagian besar instruksi-instruksinya dieksekusi dalam satu clock cycle. Dengan menggunakan prosesor ini diharapkan hasil proses akan real-time.

D. TUJUAN : Mempelajari implementasi prosesor sinyal digital TMS32010 untuk pengaturan posisi dengan menggunakan motor DC.

E. PENELAAHAN STUDI : Mempelajari sifat-sifat dan cara kerja prosesor TMS32010, meliputi arsitektur internal, organisasi hardware, bahasa assembly dan teknik-teknik interfacing.

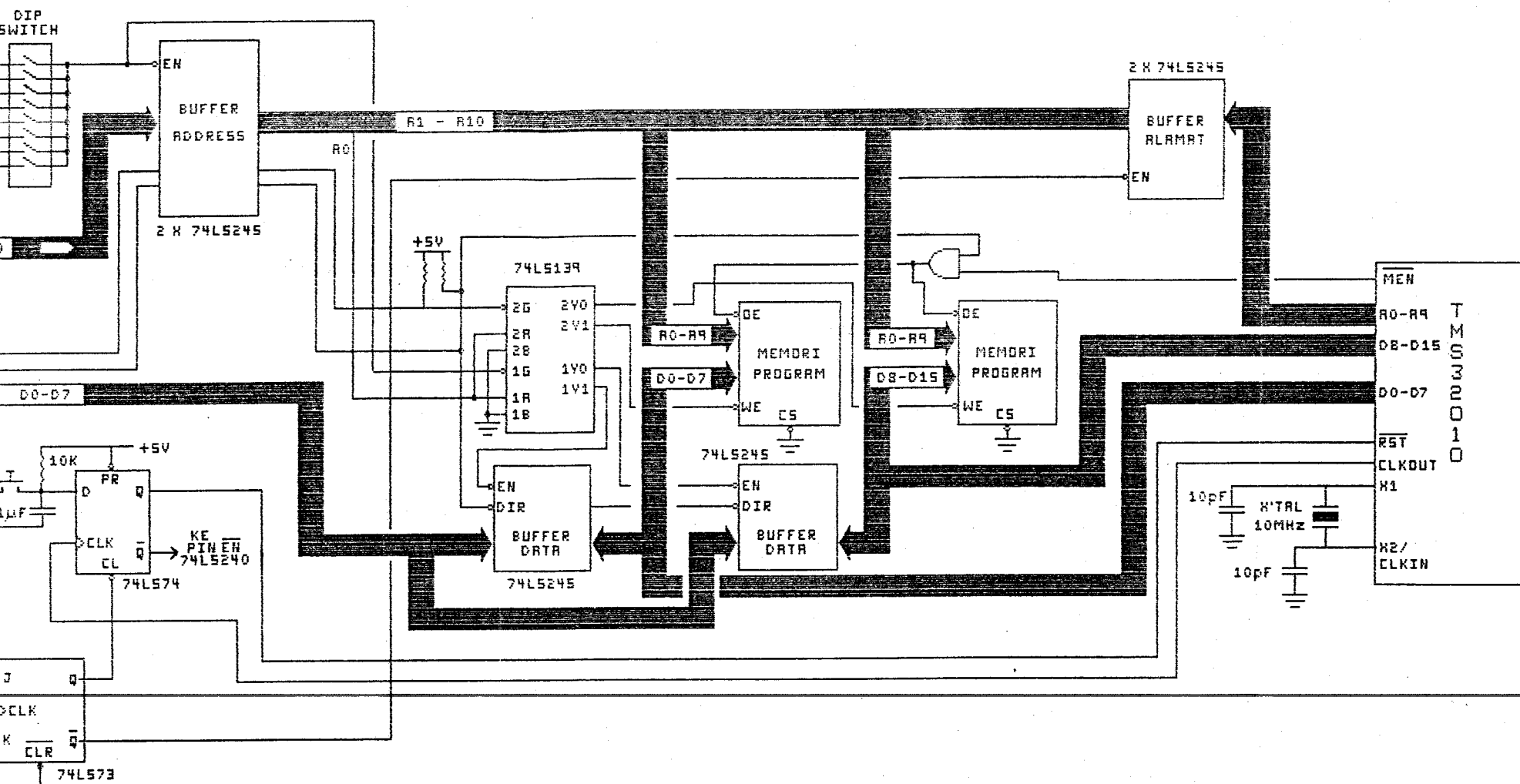
Mempelajari aplikasi prosesor sinyal digital TMS32010 sebagai pengatur proses.

F. LANGKAH-LANGKAH : 1. Studi literatur, yaitu melakukan seperti telah disebutkan di atas.  
2. Perencanaan hardware.  
3. Realisasi alat.  
4. Uji peralatan dan perbaikan.  
5. Penulisan naskah tugas akhir.

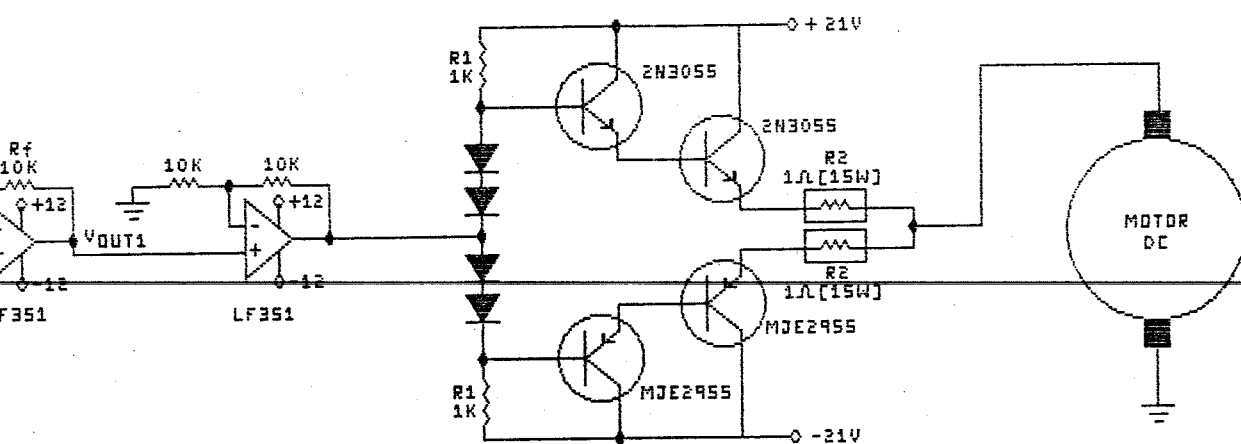
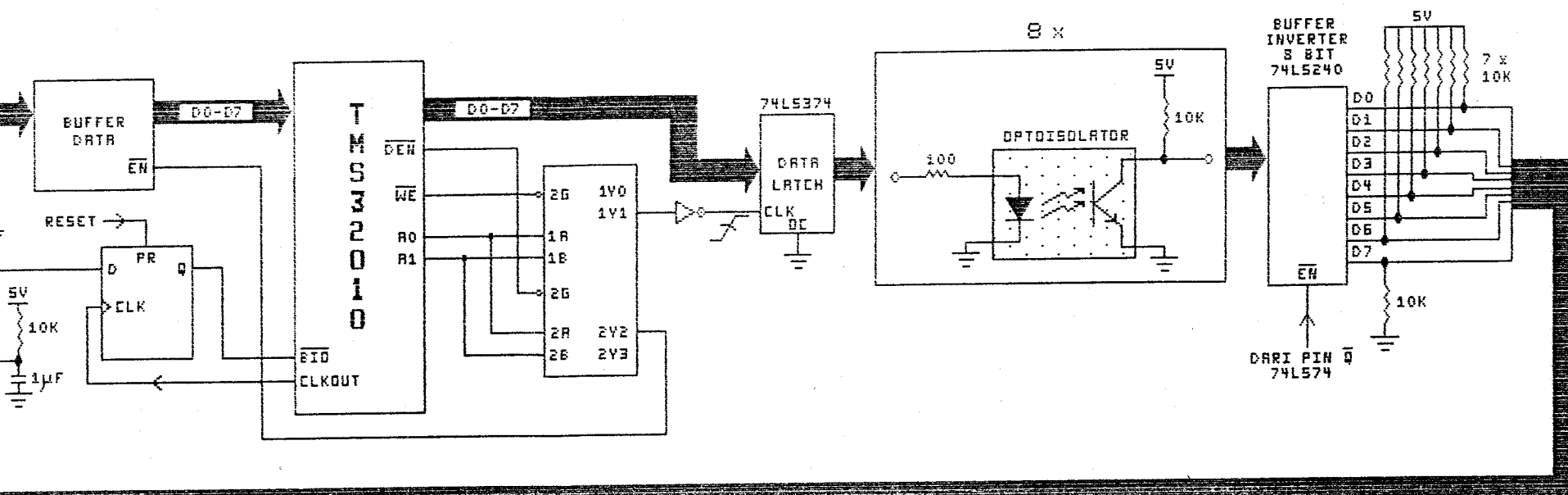
G. RELEVANSI : Prosesor sinyal digital TMS32010 bersama perangkat penunjang lainnya dapat menggantikan fungsi kontroler PID (Proportional plus Integral plus Derivative). Adapun konstanta penguatan yang diperlukan dapat diubah-ubah melalui pengaturan secara software dan memasukkan datanya melalui keyboard.

H. JADWAL KEGIATAN :

No.	KEGIATAN	BULAN KE					
		I	II	III	IV	V	VI
1.	Studi literatur	■	■	■			
2.	Perencanaan Hardware dan pengumpulan komponen		■	■	■		
3.	Realisasi alat, uji peralatan dan perbaikan			■	■	■	
4.	Penulisan naskah					■	■



RANGKAIAN INTERFACE DAN MEMORI PROGRAM TMS32010



RANGKAIAN INPUT OUTPUT